NRL Report 8074

SSPARAMA: A Nonlinear, Wave Optics Multipulse (and CW) Steady-State Propagation Code with Adaptive Coordinates

K. G. WHITNEY AND G. L. MADER

Science Applications Inc.
Arlington, Virginia

and

P. B. ULRICH

Optical Radiation Branch
Optical Sciences Division

February 10, 1977

NAVAL RESEARCH LABORATORY
Washington, D.C.

Approved for public release; distribution unlimited.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

UNCLASSIFIED

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NRL Report 8074	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) SSPARAMA: A NONLINEAR, WAVE OPTICS MULTIPULSE (AND CW) STEADY-STATE PROPAGATION CODE WITH ADAPTIVE COORDINATES		5. TYPE OF REPORT & PERIOD COVERED Final report on one phase of a continuing NRL problem 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) K. G. Whitney, G. L. Mader, and P. B. Ulrich		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Research Laboratory Washington, D.C. 20375		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS NRL Problem R05-31 Project SAA-76
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Sea Systems Command (PMS-405) Washington, D.C. 20362		12. REPORT DATE February 10, 1977 13. NUMBER OF PAGES 80
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Laser propagation
Nonlinear optics
Atmospheric optics
Computer simulation

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report describes the numerical procedures that are used to calculate the steady-state atmospheric propagation and thermal blooming of either a train of high-energy laser pulses or a CW beam. The calculation is performed through the aid of a sequence of coordinate and variable transformations that enable one to adapt the scale and location of the transverse coordinate system to the amount of diffraction or thermal blooming taking place during beam propagation. A description is also given of the input parameters and diagnostics provided in the calculation along with a listing of the computer program.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

CONTENTS

SSPARAMA: A NONLINEAR, WAVE OPTICS MULTIPULSE (AND CW) STEADY-STATE PROPAGATION CODE WITH ADAPTIVE COORDINATES

INTRODUCTION

Several methods of propagating CW high-energy laser beams through the atmosphere have been reported previously [1,2]. This report will describe a method for propagating multiply pulsed laser beams in a nonlinear atmosphere by adapting the coordinate system to the amount of thermal blooming. This technique increases the accuracy of thermal-blooming calculations and extends the capability of the code in the case of extreme beam distortion.

The computer code SSPARAMA calculates the steady-state intensity pattern of a train of high-energy laser pulses propagating through the atmosphere in the presence of thermal blooming. Steady state is achieved when enough equally spaced, equal-energy pulses have been propagated for transients in air heating to have died out. In the steady state a single pulse will propagate in an atmosphere that has been heated by many preceding pulses which have the same energy distribution as the pulse one is calculating. The pulse widths are assumed to be short compared to the sound transit time across the face of the beam, so that self-blooming will not take place. Blooming occurs only as a result of air heating by preceding pulses. However, to avoid problems of plasma formation, the pulse width must be sufficiently long that the critical intensity for air breakdown is not exceeded. Finally, as the pulse is propagated from one coordinate plane to another, coordinate transformations are performed to insure that the transverse scale lengths are adapted to the amount of thermal blooming induced on the pulse train by the negative lensing influence of the heated atmosphere.

Another requirement for steady-state propagation is that a cooling mechanism exist for removing heated air from the path of the beam. In SSPARAMA, cooling is provided either by a wind moving perpendicular to the propagation direction or by beam sluing about an axis in the aperture plane perpendicular to both the wind and the propagation directions. The steady-state density changes $\Delta\rho$ introduced in the path of a given pulse by energy absorption from all preceding pulses can then be expressed as [3]

$$\Delta\rho = -\frac{\gamma - 1}{c_s^2}\,\alpha\,E_p\,e^{-\alpha z}\sum_{n=1}^{\infty}\left|\phi(x - n\Delta t_s(v_0 + \Omega z), y, z)\right|^2, \tag{1}$$

Manuscript submitted October 14, 1976.

where

z = the distance in the propagation direction measured from the aperture plane,

x = the distance in the wind direction measured from beam maximum intensity in the aperture plane,

γ = the ratio of atmospheric specific heats (≈ 1.4),

c_s = the speed of sound in air (≈ 340 m/s),

α = the absorption coefficient for the laser radiation,

Δt_s = the pulse spacing,

E_p = the energy of each laser pulse,

v_0 = the wind speed along the x direction perpendicular to the direction of propagation, and

Ω = the angular sluing rate of the beam about the y axis.

Finally ϕ is the normalized steady-state energy distribution of each pulse at the z plane:

$$\int_{-\infty}^{\infty} |\phi(x, y, z)|^2 \, dxdy = 1. \tag{2}$$

This density reduction $\Delta\rho$ changes the index of refraction from its ambient value n_0, where $n_0 \approx 1$, to

$$n^2 \approx n_0^2 + 3N\Delta\rho,$$

where N is the molecular refractivity of air (≈ 0.154 cm^3/g). The distribution ϕ must then be calculated self-consistently from the propagation equation:

$$\left[2ik \frac{\partial}{\partial z} + \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + 3Nk^2 \Delta\rho(|\phi|^2)\right]\phi = 0, \tag{3}$$

where $k = 2\pi/\lambda$ is the wavenumber of the laser radiation. It is assumed in SSPARAMA that at $z = 0$ the pulse train has a spherical phase front and a truncated intensity profile. For example, when truncated Gaussian pulses are propagated

$$\begin{aligned} \phi(x, y, 0) &= N_g \phi_g(x, y), \qquad & x^2 + y^2 \leqslant 2a^2, \\ &= 0, & x^2 + y^2 > 2a^2, \end{aligned} \tag{4}$$

where

$$\phi_g(x, y) = \frac{1}{a\sqrt{\pi}} e^{-[1+(ika^2/f)][(x^2+y^2)/a^2]/2} \tag{5}$$

and N_g is a normalization constant insuring that Eq. (2) is satisfied at $z = 0$. Two scale lengths, a and f, are defined in Eq. (5). The scale length f, the initial curvature of the phase front, defines the distance from the aperture to the focal plane. At a distance a from the aperture center the beam intensity falls to $1/e$ of its maximum value, and the beam is truncated at $1/e^2$ of maximum intensity.

Altogether eight variable physical quantities, a, f, k, α, E_p, Δt_s, v_0, and Ω appear in Eqs. (1) through (5). All variations will not however lead to a mathematically distinct problem. In SSPARAMA Eqs. (1) through (5) are scaled so that distinct propagation problems are defined in terms of five dimensionless parameters. The program is designed to accept either the set of data with dimensions or the dimensionless set, and both sets are printed out.

The scaling of Eqs. (1) through (5) is carried out via the coordinate transformations

$$\tilde{x} \equiv \frac{x}{a}, \qquad \tilde{y} \equiv \frac{y}{a}, \qquad \tilde{z} \equiv \frac{z}{f} \tag{6}$$

and the variable transformation

$$\tilde{\phi}(\tilde{x}, \tilde{y}, \tilde{z}) \equiv a\phi(x, y, z). \tag{7}$$

By multiplying Eq. (3) through by a^3, one can write the propagation equation in a form which identifies the five dimensionless parameters characterizing propagation in SSPARAMA:

$$\left\{2iN_k \frac{\partial}{\partial \tilde{z}} + \frac{\partial^2}{\partial \tilde{x}^2} + \frac{\partial^2}{\partial \tilde{y}^2} - N_k N_c e^{-N_\alpha z} \sum_{n=1}^{\infty} \left| \tilde{\phi}\left[\tilde{x} - \frac{2n}{N_o}(1 + N_s \tilde{z}), \tilde{y}, \tilde{z}\right]\right|^2 \right\} \tilde{\phi} = 0. \tag{8}$$

The five parameters, N_k, N_c, N_α, N_o, and N_s, are defined as

$$N_k = ka^2/f, \tag{9}$$

$$N_c = \frac{3Nk(\gamma - 1)\alpha f E_p}{c_s{}^2 a^2}, \tag{10}$$

$$N_\alpha = \alpha f, \tag{11}$$

$$N_o = \frac{2a}{v_0 \Delta t_s}, \tag{12}$$

and

$$N_s = \Omega f / v_0. \tag{13}$$

N_k is the Fresnel number of the free-propagation problem, and N_c, N_α, N_o, and N_s are coupling strength, absorption, overlap, and sluing parameters respectively. N_o was introduced by Wallace and Lilly [4] and called the pulses-per-flow-time parameter. It measures the number of preceding pulses which have heated the air across the beam aperture as the pulse under study begins to propagate. The solution to Eq. (8) is obtained subject to the energy normalization

$$\int |\tilde{\phi}(\tilde{x}, \tilde{y}, 0)|^2 \, d\tilde{x} \, d\tilde{y} = 1 \tag{14}$$

and the initial condition

$$\tilde{\phi}(\tilde{x}, \tilde{y}, 0) = |\tilde{\phi}| \, e^{-iN_k(\tilde{x}^2+\tilde{y}^2)/2}, \tag{15}$$

where $|\tilde{\phi}| = 0$ for $\tilde{x}^2 + \tilde{y}^2 > 2$.

Equations (8), (14), and (15) are numerically solved in SSPARAMA on a 64-by-64 grid in the $\tilde{x}\tilde{y}$ plane. Since one would like to use as much of the computational grid as possible to describe the variations in beam intensity, a scheme for adapting the coordinate grid to the propagation must be used. For example, as the beam propagates, the initial focusing causes the beam intensity pattern to decrease in size until the negative lensing effects of the heated atmosphere accumulate to thermally defocus it. Moreover, since the wind removes heated air from the path of the beam from left to right, a thermal gradient is established that deflects the beam from right to left. If the computational grid were not moved or changed in size as the beam intensity was calculated from aperture to focal plane, the intensity pattern would either be poorly sampled as it decreased in size or it would expand or deflect to reach the boundary of the grid and invalidate the calculation.

A technique for adapting the computational grid to local changes in the size or location of the beam intensity pattern has been developed by Herrmann and Bradley [5]. A slightly modified form of their technique has been incorporated into SSPARAMA and will be described in the next section of this report. In the third section the numerical procedures used in SSPARAMA will be described, and in the fourth section the code usage will be explained.

COORDINATE-SYSTEM ADAPTION

The dimensionless form of the propagation equation can be rewritten more compactly as

$$[2iN_k \partial_{\tilde{z}} + \partial_{\tilde{x}}^2 + \partial_{\tilde{y}}^2 + k^2a^2(n^2 - 1)]\tilde{\phi} = 0, \tag{16}$$

where $n^2 - 1$, the nonlinear index of refraction, depends on $\tilde{\phi}$ as given by Eq. (8). The $\tilde{x}\tilde{y}\tilde{z}$ coordinate system is normalized to the constant lengths a and f, and is fixed in space. In this system therefore the beam will lie symmetrically about the origin of the $\tilde{x}\tilde{y}$ plane only at $\tilde{z} = 0$ with an extent of order 1 (see, for example, Eq. (15)). When $z \neq 0$, a new set of xy coordinates is needed to maintain the two properties that the beam be centered about the xy coordinate origin and be of order 1 in extent. In general, one can relate the xy and $\tilde{x}\tilde{y}$ coordinates by a set of scale parameters D_1 and D_2 and a deflection parameter X, which are functions of $\tilde{z}$. Since one would like to solve Eq. (16) in a set of coordinates that adapt to changes in beam size and direction, the coordinate transformation must be related to these beam changes as determined by the linear and quadratic terms of the phase front. By analogy therefore with the transformation to dimensionless parameters, one must perform simultaneous coordinate and variable transformations. The form of these transformations is suggested by linear propagation theory:

$$x = \frac{\tilde{x} - X}{\sqrt{D_1}}, \tag{17}$$

$$y = \frac{\tilde{y}}{\sqrt{D_2}}, \tag{18}$$

$$z = \frac{\tilde{z}}{N_k}, \tag{19}$$

and

$$\tilde{\phi} = \frac{\psi}{\sqrt[4]{D_1 D_2}} \, e^{i(\tilde{\alpha}_1 \tilde{x}^2 + \tilde{\alpha}_2 \tilde{y}^2 + \tilde{\beta}\tilde{x} + \tilde{\gamma}_1 + \tilde{\gamma}_2)}. \tag{20}$$

The constant scale change from $\tilde{z}$ to z is done for convenience to eliminate N_k from the z-derivative term in Eq. (16):

$$2iN_k \partial_{\tilde{z}} \rightarrow 2i\partial_z.$$

The factor $1/\sqrt[4]{D_1 D_2}$ is removed from $\tilde{\phi}$ to insure the form invariance of the energy normalization:

$$\int |\tilde{\phi}|^2 \, d\tilde{x} d\tilde{y} = \int |\psi|^2 \, dx dy = 1. \tag{21}$$

When Eqs. (17) through (20) are substituted into Eq. (16) and when the nonlinear term is of negligible size and the beam has a Gaussian profile, D_1, D_2, X, $\tilde{\alpha}_1$, $\tilde{\alpha}_2$, $\tilde{\beta}$, $\tilde{\gamma}_1$, and $\tilde{\gamma}_2$ as functions of z can be analytically determined for all z. However, when the nonlinear term is important or when a non-Gaussian beam is propagated, the $\tilde{\alpha}$'s and $\tilde{\beta}$, which represent the effective quadratic and linear phase changes throughout the xy plane, can no longer be so determined. One must adopt a more limited strategy for the employment of Eqs. (17) through (20).

Consider, for example, that the quantities D_1, D_2, X, $\tilde{\alpha}_1$, $\tilde{\alpha}_2$, $\tilde{\beta}$, $\tilde{\gamma}_1$, and $\tilde{\gamma}_2$ are known at $z = z_0$ and that their dependence on z is to be analytically determined as one propagates to a neighboring xy plane at $z_0 + \Delta z$. Since

$$\partial_{\tilde{x}}^2 = \frac{1}{D_1} \partial_x^2, \tag{22}$$

$$\partial_{\tilde{y}}^2 = \frac{1}{D_2} \partial_y^2, \tag{23}$$

and

$$\partial_{\tilde{z}} = \frac{1}{N_k}\left[\partial_z - \left(\frac{x}{2}\partial_z \ln D_1 + \frac{\partial_z X}{\sqrt{D_1}}\right)\partial_x - \frac{y}{2}\partial_z \ln D_2 \,\partial_y\right], \tag{24}$$

one finds that

$$[2iN_k\partial_{\tilde{z}} + \partial_{\tilde{x}}^2 + \partial_{\tilde{y}}^2 + k^2a^2(n^2-1)]\frac{\psi}{\sqrt[4]{D_1D_2}}\; e^{i(\tilde{\alpha}_1\tilde{x}^2+\tilde{\alpha}_2\tilde{y}^2+\tilde{\beta}\tilde{x}+\tilde{\gamma}_1+\tilde{\gamma}_2)}$$

$$= \frac{e^{i(\tilde{\alpha}_1\tilde{x}^2+\tilde{\alpha}_2\tilde{y}^2+\tilde{\beta}\tilde{x}+\tilde{\gamma}_1+\tilde{\gamma}_2)}}{\sqrt[4]{D_1D_2}}\left\{2i\left(\partial_z - \frac{x}{2}\partial_z \ln D_1\,\partial_x - \frac{1}{\sqrt{D_1}}\partial_z X\partial_x - \frac{y}{2}\partial_z \ln D_2\,\partial_y\right)\right.$$

$$- \frac{i}{2}(\partial_z \ln D_1 + \partial_z \ln D_2) - 2\partial_z(\tilde{\gamma}_1+\tilde{\gamma}_2) + \frac{1}{D_1}\partial_x^2 - [2\tilde{\alpha}_1(\sqrt{D_1}\,x + X) + \tilde{\beta}]^2$$

$$+ \frac{2i}{\sqrt{D_1}}[2\tilde{\alpha}_1(\sqrt{D_1}\,x + X) + \tilde{\beta}]\partial_x + 2i\tilde{\alpha}_1 + \frac{1}{D_2}\partial_y^2 - 4\tilde{\alpha}_2^2 D_2 y^2$$

$$\left. + 4i\tilde{\alpha}_2 y\partial_y + 2i\tilde{\alpha}_2 + k^2a^2(n^2-1)\right\}\psi = 0. \tag{25}$$

For vanishingly small $n^2 - 1$ and for a real Gaussian profile $\psi(x, y, z_0)$ one would determine D_1, D_2, X, $\tilde{\alpha}_1$, $\tilde{\alpha}_2$, $\tilde{\beta}$, $\tilde{\gamma}_1$, and $\tilde{\gamma}_2$ from the requirement that Eq. (25) be capable of being put in the form

$$\left[2i\,\partial_z + \frac{1}{D_1}(\partial_x^2 + 1 - x^2) + \frac{1}{D_2}(\partial_y^2 + 1 - y^2) + k^2a^2(n^2-1)\right]\psi = 0. \tag{26}$$

Then, as ψ was propagated to $z_0 + \Delta z$, it would acquire no z dependence and would remain real and Gaussian; that is, all of the z dependence of ϕ would have been accounted for in $D_1, \cdots, \tilde{\gamma}_2$.

For the imaginary terms of Eq. (25) other than $2i\,\partial_z$ to vanish, the quantities D_1, D_2, and X, which determine the scale and location of the xyz coordinate system, must satisfy the equations

$$\partial_z \ln D_1 = 4\tilde{\alpha}_1, \tag{27}$$

$$\partial_z \ln D_2 = 4\tilde{\alpha}_2, \tag{28}$$

and

$$\partial_z X = 2\tilde{\alpha}_1 X + \tilde{\beta}. \tag{29}$$

On the other hand, for the real terms involving ∂_x and ∂_y to vanish and for the scale functions D_1 and D_2 to be factorable from the remaining x and y terms respectively, the phase functions $\tilde{\alpha}_1$, $\tilde{\alpha}_2$, $\tilde{\beta}$, $\tilde{\gamma}_1$, and $\tilde{\gamma}_2$ must satisfy the set of equations

$$2D_1\partial_z\tilde{\alpha}_1 + 4\tilde{\alpha}_1^2 D_1 = \frac{1}{D_1}, \tag{30}$$

$$2D_2\partial_z\tilde{\alpha}_2 + 4\tilde{\alpha}_2^2 D_2 = \frac{1}{D_2}, \tag{31}$$

$$\partial_z\tilde{\beta} + 2X\partial_z\tilde{\alpha}_1 + 2\tilde{\alpha}_1(2\tilde{\alpha}_1 X + \tilde{\beta}) = 0, \tag{32}$$

$$2\partial_z\tilde{\gamma}_1 + 2X^2\partial_z\tilde{\alpha}_1 + 2X\partial_z\tilde{\beta} + (2\tilde{\alpha}_1 X + \tilde{\beta})^2 = -\frac{1}{D_1}, \tag{33}$$

and

$$2\partial_z\tilde{\gamma}_2 = -\frac{1}{D_2}. \tag{34}$$

Thus Eqs. (27) through (34) will determine all of the z dependence of $\tilde{\phi}$ when $\psi(x, y, z_0)$ is real and a Gaussian function of x and y and there is no lensing effect caused by heating of the atmosphere; that is, Eqs. (27) through (34) will describe beam focusing in the absence of diffraction and nonlinear media phenomena. They are of more limited utility when such phenomena are present. In this case, during the displacement of ϕ from z_0 to $z_0 + \Delta z$, linear and quadratic phase changes will arise from two sources. As a result of focusing at $z = z_0$, the initial phases $\tilde{\alpha}_1(z_0)$, $\tilde{\alpha}_2(z_0)$, and $\tilde{\beta}(z_0)$ will become $\tilde{\alpha}_1(z_0 + \Delta z)$, $\tilde{\alpha}_2(z_0 + \Delta z)$, and $\tilde{\beta}(z_0 + \Delta z)$ through the solution to Eqs. (27) through (34). In addition however ψ at $z_0 + \Delta z$ will acquire linear and quadratic phases, $\Delta\tilde{\beta}$, $\Delta\tilde{\alpha}_1$, and $\Delta\tilde{\alpha}_2$ respectively, as a result of diffraction and thermal blooming. Thus at $z_0 + \Delta z$ a new factorization of $\tilde{\phi}$ must be made, namely,

$$\tilde{\phi}(\tilde{x}, \tilde{y}, \tilde{z}_0 + \Delta\tilde{z}) \equiv \frac{\psi'(x, y, z_0 + \Delta z)}{\sqrt[4]{D_1(z_0 + \Delta z)D_2(z_0 + \Delta z)}}\, e^{i[\tilde{\alpha}_1'(z_0+\Delta z)\tilde{x}^2 + \tilde{\alpha}_2'(z_0+\Delta z)\tilde{y}^2 + \tilde{\beta}'(z_0+\Delta z)\tilde{x} + \tilde{\gamma}_1' + \tilde{\gamma}_2']}, \tag{35}$$

if ψ', which is to be propagated from $z_0 + \Delta z$ to $z_0 + \Delta z + \Delta z'$, is not to initially have quadratic or linear phase terms. After each step in propagation therefore $\tilde{\alpha}_1$, $\tilde{\alpha}_2$, and $\tilde{\beta}$ must be redefined as

$$\tilde{\alpha}_1'(z_0 + \Delta z) = \tilde{\alpha}_1(z_0 + \Delta z) + \Delta\tilde{\alpha}_1, \tag{36}$$

$$\tilde{\alpha}_2'(z_0 + \Delta z) = \tilde{\alpha}_2(z_0 + \Delta z) + \Delta\tilde{\alpha}_2, \tag{37}$$

and

$$\tilde{\beta}'(z_0 + \Delta z) = \tilde{\beta}(z_0 + \Delta z) + \Delta\tilde{\beta} \tag{38}$$

in order to adapt the coordinate-system determination from Eqs. (27) through (29) to changes in phase that result from focusing, diffraction, and thermal blooming.

In SSPARAMA, ψ is propagated from one z plane to another by finite-differencing a phase-transformed version of Eq. (26). Then $\Delta\alpha_1$, $\Delta\alpha_2$, and $\Delta\beta$ are *found in the xyz coordinate system* using the method of phase minimization discussed by Herrmann and Bradley [5]. One requires that

$$\int_{z=z_0+\Delta z} |\psi|^2[\nabla(\Delta\alpha_1 x^2 + \Delta\alpha_2 y^2 + \Delta\beta x - \gamma)]^2\, dxdy = \text{minimum}, \tag{39}$$

where $\psi(x, y, z_0 + \Delta z) \equiv |\psi| e^{i\gamma}$. It follows that

$$\Delta\alpha_1 = \frac{D_1 E - B_1 C_1}{2(A_1 E - B_1^2)}, \tag{40}$$

$$\Delta\beta = \frac{A_1 C_1 - B_1 D_1}{A_1 E - B_1^2}, \tag{41}$$

and

$$\Delta\alpha_2 = \frac{D_2}{2A_2}, \tag{42}$$

where

$$A_1 \equiv \int x^2 |\psi|^2\, dxdy, \quad A_2 \equiv \int y^2 |\psi|^2\, dxdy, \tag{43}$$

$$B_1 \equiv \int x |\psi|^2\, dxdy, \tag{44}$$

$$C_1 \equiv \text{Im} \int \psi^* \partial_x \psi\, dxdy, \tag{45}$$

$$D_1 \equiv \text{Im} \int x\psi^* \partial_x \psi\, dxdy, \quad D_2 \equiv \text{Im} \int y\psi^* \partial_y \psi\, dxdy, \tag{46}$$

and

$$E \equiv \int |\psi|^2\, dxdy = 1. \tag{47}$$

The factorization

$$\psi(x, y, z_0 + \Delta z) \equiv \psi' e^{i(\Delta\alpha_1 x^2 + \Delta\alpha_2 y^2 + \Delta\beta x)} \tag{48}$$

will then define ψ' at $z_0 + \Delta z$ as a wave function of minimum quadratic and linear phase. In particular, if ψ is exactly a Gaussian beam, ψ' will be real.

The relationship between $\{\Delta\alpha_1, \Delta\alpha_2, \Delta\beta\}$ and $\{\Delta\tilde{\alpha}_1, \Delta\tilde{\alpha}_2, \Delta\tilde{\beta}\}$ is found by substituting Eqs. (17) and (18) into Eq. (48):

$$\Delta\tilde{\alpha}_1 = \frac{\Delta\alpha_1}{D_1}, \tag{49}$$

$$\Delta\tilde{\alpha}_2 = \frac{\Delta\alpha_2}{D_2}, \tag{50}$$

and

$$\Delta\tilde{\beta} = \frac{\Delta\beta}{\sqrt{D_1}} - \frac{2\Delta\alpha_1 X}{D_1}. \tag{51}$$

A similar set of equations will hold between $\{\tilde{\alpha}_1, \tilde{\alpha}_2, \tilde{\beta}\}$ and $\{\alpha_1, \alpha_2, \beta\}$, which are computed directly in the *xyz* coordinate system. When reexpressed in terms of α_1, α_2 and β, Eqs. (27) through (29) become

$$\partial_z D_1 = 4\alpha_1, \tag{52}$$

$$\partial_z D_2 = 4\alpha_2, \tag{53}$$

and

$$\partial_z X = \frac{\beta}{\sqrt{D_1}}, \tag{54}$$

and Eqs. (30) through (32) transform into

$$\partial_z \alpha_1 = \frac{1}{2D_1}(1 + 4\alpha_1^2), \tag{55}$$

$$\partial_z \alpha_2 = \frac{1}{2D_2}(1 + 4\alpha_2^2), \tag{56}$$

and

$$\partial_z \beta = \frac{2\alpha_1\beta}{D_1}. \tag{57}$$

Eqs. (52) through (57) must be solved in terms of initial values at z_0. The solutions are

$$D_{1,2}(z) = D_{1,2}(z_0)\left\{\left[1 + \frac{2\alpha_{1,2}(z_0)}{D_{1,2}(z_0)}(z - z_0)\right]^2 + \left[\frac{z - z_0}{D_{1,2}(z_0)}\right]^2\right\}, \tag{58}$$

$$\alpha_{1,2}(z) = \alpha_{1,2}(z_0) + \frac{1}{2}\{1 + [2\alpha_{1,2}(z_0)]^2\}\frac{z - z_0}{D_{1,2}(z_0)}, \tag{59}$$

$$\beta(z) = \beta(z_0)\sqrt{\left[1 + \frac{2\alpha_1(z_0)}{D_1(z_0)}(z - z_0)\right]^2 + \left[\frac{z - z_0}{D_1(z_0)}\right]^2}, \tag{60}$$

and

$$X(z) = X(z_0) + \frac{\beta(z_0)}{\sqrt{D_1(z_0)}}(z - z_0). \tag{61}$$

Finally the procedure for solving Eq. (26) in SSPARAMA is similar to the one described in an earlier report [2]. A phase transformation on ψ is made:

$$\Phi(x, y, z) \equiv \psi(x, y, z)\, e^{-(i/2)\int_{z_0'}^{z} g(x,y,z')dz'}, \tag{62}$$

where

$$g(x, y, z) \equiv \frac{1}{D_1}(1 - x^2) + \frac{1}{D_2}(1 - y^2) + k^2a^2(n^2 - 1). \tag{63}$$

The equation for Φ follows from Eq. (26):

$$[2i\,\partial_z + H(x, y, z)]\Phi = 0, \tag{64}$$

where

$$H = e^{-(i/2)\int_{z_0'}^{z} g\,dz'}\left(\frac{1}{D_1}\partial_x^{\,2} + \frac{1}{D_2}\partial_y^{\,2}\right)e^{(i/2)\int_{z_0'}^{z} g\,dz'}. \tag{65}$$

By picking z_0' to lie between z_0 and $z_0 + \Delta z$, one can propagate Φ from z_0 to $z_0 + \Delta z_0$, with first-order accuracy, by solving the equation

$$[2i\partial_z + H(x, y, z_0')]\Phi = \left(2i\partial_z + \frac{1}{D_1}\partial_x^{\,2} + \frac{1}{D_2}\partial_y^{\,2}\right)\Phi = 0. \tag{66}$$

Equation (66) is solved by Fourier transforming Φ [6],

$$\tilde{\Phi}(k_1, k_2, z_0) \equiv \int e^{i(k_1x+k_2y)}\,\Phi(x, y, z_0)\,dxdy, \tag{67}$$

and propagating $\tilde{\Phi}$ to $z_0 + \Delta z$:

$$\tilde{\Phi}(k_1, k_2, z_0 + \Delta z) = \tilde{\Phi}(k_1, k_2, z_0)\,e^{(i/2)\{k_1^2\int_{z_0}^{z_0+\Delta z}[1/D_1(z)]dz + k_2^2\int_{z_0}^{z_0+\Delta z}[1/D_2(z)]dz\}}. \tag{68}$$

The inverse transformation to Eq. (67) then yields Φ, and Eq. (62) yields $\psi(x, y, z_0 + \Delta z)$.

NUMERICAL PROCEDURES

The phase function $g(x, y, z)$ of Eq. (63) can be written more usefully in the form

$$g = \frac{g_1(x)}{D_1(z)} + \frac{g_2(y)}{D_2(z)} - \frac{g_3(x, y, z)}{\sqrt{D_1(z)D_2(z)}}, \tag{69}$$

where

$$g_1(x) \equiv 1 - x^2, \tag{70}$$

$$g_2(y) \equiv 1 - y^2, \tag{71}$$

and

$$g_3(x, y, z) \equiv N_k N_c e^{-N_\alpha N_k z} \sum_{n=1}^{\infty} \left| \Phi \left[x - \frac{2n}{N_0\sqrt{D_1(z)}}(1 + N_s N_k z), y, z \right] \right|^2. \tag{72}$$

This expression for g_3 is found by substituting the new variables x, y, z, and Φ into Eq. (8). The phase integral

$$\Delta\theta \equiv \int_{z_0'}^{z} g(x, y, z')\, dz'$$

appearing in Eq. (62) can now be partially evaluated and expressed in the form

$$\Delta\theta = g_1(x)\Delta Z_1 + g_2(y)\Delta Z_2 - \int_{z_0'}^{z} \frac{g_3(x, y, z)}{\sqrt{D_1(z)D_2(z)}} dz, \tag{73}$$

where

$$\Delta Z_{1,2} \equiv \int_{z_0'}^{z} \frac{dz'}{D_{1,2}(z')} = \tanh^{-1}\left(\{1 + [2\alpha_{1,2}(z_0)]^2\} \frac{z' - z_0}{D_{1,2}(z_0)} + 2\alpha_{1,2}(z_0)\right)\Bigg|_{z'=z_0'}^{z'=z}. \tag{74}$$

The differential quantities ΔZ_1 and ΔZ_2 are similarly named as the coordinate differential ΔZ that was used in earlier code calculations which involved only a single scaling function $D(z)$.

To complete the evaluation of $\Delta\theta$, one must know the z dependence of g_3, that is, the z dependence of $|\Phi|^2$. Two options are provided in SSPARAMA, for evaluating $\Delta\theta$, depending on whether one has determined $|\Phi|^2$ at one or both of the integration endpoints. The procedures work as follows: Suppose first that the solution for $\psi(x, y, z_0)$ has been obtained. Then one can compute $g(x, y, z_0)$, since $|\Phi(x, y, z_0)|^2 = |\psi(x, y, z_0)|^2$. To find $\Phi(x, y, z_0)$, however, one must evaluate

$$\Delta\theta' \equiv \int_{z_0}^{z_0'} g(x, y, z')\, dz', \tag{75}$$

where z_0' lies between z_0 and the plane $z_0 + \Delta z$ to which one would like to propagate ψ. If ψ is known only at z_0, the zeroth-order approximation

$$\Delta\theta' \approx g_1(x)\Delta Z_1' + g_2(y)\Delta Z_2' - g_3(x, y, z_0)\Delta Z_{12}' \tag{76}$$

must be made, where

$$\Delta Z_{12}' \equiv \int_{z_0}^{z_0'} \frac{dz'}{\sqrt{D_1(z')D_2(z')}}. \tag{77}$$

Equation (66) can now be solved for $\Phi(x, y, z_0 + \Delta z)$ by the use of Fourier transformations. Finally on performance of the phase integral

$$\Delta\theta'' \equiv \int_{z_0'}^{z_0+\Delta z} g(x, y, z')\, dz' \tag{78}$$

$\psi(x, y, z_0 + \Delta z)$ can be obtained from $\Phi(x, y, z_0 + \Delta z)$. In keeping with the accuracy with which $\Delta\theta'$ was approximated, $\Delta\theta''$ can be approximately evaluated as

$$\Delta\theta'' \approx g_1(x)\Delta Z_1'' + g_2(y)\Delta Z_2'' - g_3(x, y, z_0 + \Delta z)\Delta Z_{12}''. \tag{79}$$

The differentials $\Delta Z_1''$, $\Delta Z_2''$, and $\Delta Z_{12}''$ are defined by the integrals of Eqs. (74) and (77) with the integration limits as specified in Eq. (78).

Suppose however that initially both $\psi(x, y, z_0)$ and $\psi(x, y, z_0')$ are known and that the values of ψ at z_0 are to be propagated to the plane at $z_0 + \Delta z$. In this case the phase integrals defined in Eqs. (75) and (78) can be approximated using the integration formula

$$\int_{x_0}^{x_0+\Delta x} f(x)g(x)\, dx \approx w_1 f(x_0) + w_2 f(x_0 + \Delta x), \tag{80}$$

which has first-order instead of zeroth-order accuracy. The weights w_1 and w_2 are thus determined such that equality will hold in Eq. (80) whenever f is a linear function of x:

$$w_1 = \left(1 + \frac{2x_0}{\Delta x}\right)\int_{x_0}^{x_0+\Delta x} g(x)\, dx - \frac{2}{\Delta x}\int_{x_0}^{x_0+\Delta x} xg(x)\, dx \tag{81}$$

and

$$w_2 = \frac{2}{\Delta x}\int_{x_0}^{x_0+\Delta x} xg(x)\,dx - \frac{2x_0}{\Delta x}\int_{x_0}^{x_0+\Delta x} g(x)\,dx. \tag{82}$$

Then, for example, in place of Eq. (76) one would have that

$$\Delta\theta' \approx g_1(x)\Delta Z_1' + g_2(y)\Delta Z_2' - g_3(x, y, z_0)\Delta Z_3' - g_3(x, y, z_0')\Delta Z_4', \tag{83}$$

where $\Delta Z_3'$ and $\Delta Z_4'$ are related through Eqs. (81) and (82) to $\Delta Z_{12}'$ and an integration over the function $z/\sqrt{D_1(z)D_2(z)}$:

$$\Delta Z_3' = \frac{z_0' + z_0}{z_0' - z_0}\Delta Z_{12}' - \frac{2}{z_0' - z_0}\int_{z_0}^{z_0'} \frac{z'\,dz'}{\sqrt{D_1(z')D_2(z')}} \tag{84}$$

and

$$\Delta Z_4' = \frac{2}{z_0' - z_0}\left[\int_{z_0}^{z_0'} \frac{z'\,dz'}{\sqrt{D_1(z')D_2(z')}} - z_0\Delta Z_{12}'\right]. \tag{85}$$

Although integrations over D_1^{-1} and D_2^{-1} can be carried out analytically in terms of inverse hyperbolic tangents (as in Eq. (74)), integrals over $1/\sqrt{D_1D_2}$ produce elliptic functions. Both sets of integrations are handled in SSPARAMA numerically, with third-order accuracy, using a second integration formula:

$$\int_{x_0}^{x_0+\Delta x} f(x)\,dx \approx \frac{\Delta x}{2}\left[f(x_0 + \Delta x_1) + f(x_0 + \Delta x_2)\right], \tag{86}$$

where $\Delta x_1 \equiv (1 - 1/\sqrt{3})\Delta x/2$ and $\Delta x_2 \equiv (1 + 1/\sqrt{3})\Delta x/2$. Again, as an example, consider Eqs. (84) and (85) and define

$$f_1 \equiv \frac{1}{\sqrt{D_1(z_1)D_2(z_1)}} \tag{87}$$

and

$$f_2 \equiv \frac{1}{\sqrt{D_1(z_2)D_2(z_2)}}, \tag{88}$$

where $z_1 \equiv z_0 + (1 - 1/\sqrt{3})[(z_0' - z_0)/2]$ and $z_2 \equiv z_0 + (1 + 1/\sqrt{3})[(z_0' - z_0)/2]$. One can complete the numerical evaluation of $\Delta Z_3'$ and $\Delta Z_4'$ by rewriting Eqs. (84) and (85) with the use of Eq. (86), in terms of f_1 and f_2:

$$\Delta Z_3' = \frac{z_0' - z_0}{2\sqrt{3}}(f_1 - f_2) \tag{89}$$

and

$$\Delta Z_4' = \frac{z_0' - z_0}{2}\left[\left(1 - \frac{1}{\sqrt{3}}\right)f_1 + \left(1 + \frac{1}{\sqrt{3}}\right)f_2\right]$$

$$= (z_0' - z_0)\left(\frac{f_1 + f_2}{2}\right) - \Delta Z_3'. \tag{90}$$

The procedure by which Eqs. (80) through (90) are employed requires that two sets of values of ψ be stored at any time by SSPARAMA. At the beginning of the propagation step described above, the two arrays contain the values of $\psi(x, y, z_0)$ and $\psi(x, y, z_0')$, where $z_0 < z_0' < z_0 + \Delta z$. At the end of the propagation step the values of $\psi(x, y, z_0)$ have been replaced by $\psi(x, y, z_0 + \Delta z)$. These new values can then be used to propagate $\psi(x, y, z_0')$ to $\psi(x, y, z_0' + \Delta z')$, where now $z_0' < z_0 + \Delta z < z_0' + \Delta z'$. The process of alternatively propagating one and then the other of the two arrays is repeated until the focal plane, defined by the initial beam curvature, is reached.

Since both arrays are initially assigned the values $\psi(x, y, 0)$, the process of propagating one array past the other cannot begin until after the first propagation step. The first z step is therefore taken using Eqs. (76) and (79) to determine $\Delta\theta'$ and $\Delta\theta''$. In general the incremental steps Δz are selected in SSPARAMA according to a criterion that the phase changes induced by g_3 as computed from Eq. (76) be no larger than some preassigned value of order 1 for all x and y. However, to carry out the first advancement of ψ at $z_0 = 0$, half of the initially computed Δz value is used. This leapfrog procedure is summarized for the first few z steps in Fig. 1.

The advantage conveyed by using Eqs. (76) and (79) to evaluate the phase integrals $\Delta\theta'$ and $\Delta\theta''$ is that only one ψ array is needed in carrying out the calculation. Because of the reduced accuracy in computing $\Delta\theta'$ and $\Delta\theta''$, however, smaller z steps are in principle required to obtain the same results as when two arrays at different z planes are used. To allow a quantitative comparison of these two procedures, both options for propagating ψ were installed in SSPARAMA and can be selected according to the value of one of the input parameters to the code. For the same reason, another input parameter is also available that allows one to adapt or not adapt the coordinate system to the amount of diffraction or thermal blooming occurring during beam propagation.

PROGRAM OPERATION

This section will describe the input parameters required to run SSPARAMA and explain the data included in the output. A complete listing of SSPARAMA is included in Appendix A.

To use program SSPARAMA, two input cards are required. The first specifies certain numerical parameters and selects various program options, and the second defines the

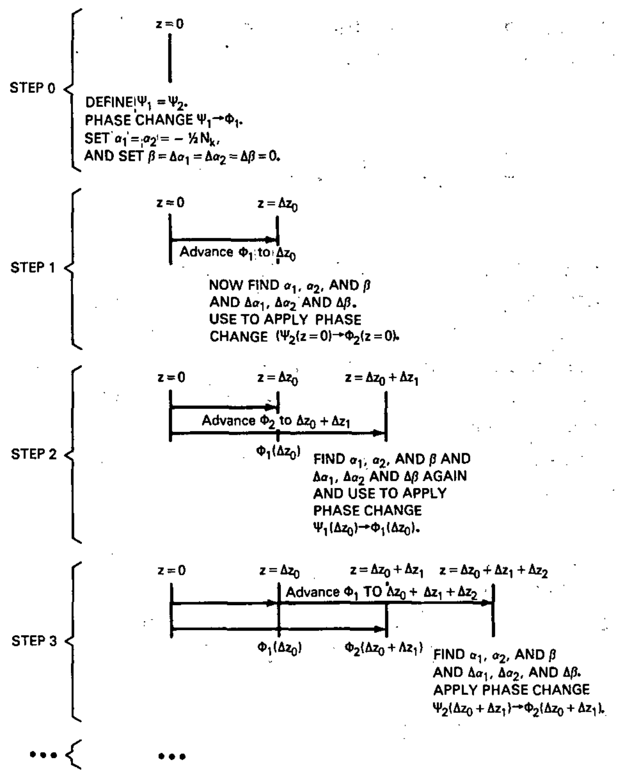

Fig. 1—Leapfrog procedure for advancing the wave function Φ

particular physical situation. This second card can contain the actual physical parameters or a set of dimensionless parameters.

First Input Card

The parameters read from the first card are listed in Table 1. A description of each of these parameters is as follows:

Table 1—Parameters Specified by the First Input Card

Columns	Name	Format	Columns	Name	Format
1-5	PHIMXX	F5.0	36-40	NPM	I5
6-10	ROCULT	F5.0	41-45	NBM	I5
11-15	HXY	F5.0	46-50	NPLOT	I5
16-20	NXY	I5	51-55	NCT	I5
21-25	NCW	I5	56-60	NRS	I5
26-30	NAD	I5	61-65	NPUNCH	I5
31-35	NMS	I5	75-80	NID	A6

PHIMXX. This is the maximum allowed phase change in radians for any point in the computational grid at each z step. It is used to define the newly computed z increments HZN at each step, where

$$\frac{g_3(x, y, z)_{\max}}{\sqrt{D_1 D_2}}\ \mathrm{HZN} = \mathrm{PHIMXX},$$

in which $g_3(x, y, z)_{\max}$ is the maximum value in the computational grid of g_3, given by Eq. (72). PHIMXX is nominally entered as 1.0. If more z steps are required, PHIMXX can be decreased. In this case the z increment is tied to the amount of heating in the atmosphere, becoming smaller automatically as large density changes take place or becoming large and efficient when near-vacuumlike propagation occurs. If HZN exceeds 0.1 of the total propagation distance, the smaller of these two z increments is used. If HZN at any time is less than 10^{-7} times the distance to be propagated, the program exits and an error message will be printed.

ROCULT. This is used when propagating uniform circular beamshapes with an obscuring disk or a uniform rectangular beamshape. In the former case ROCULT is the ratio of the occulting radius to the total radius. For a rectangle, it is the ratio of the y to the x dimension. ROCULT is used only when NBM equals 4 or 5.

HXY. This parameter defines the size of the computational grid relative to the aperture radius by

$$\Delta x = \Delta y = \mathrm{HXY}$$

where Δx and Δy are the sizes of individual computational cells, which start out square. Depending on the beamshape, values between 0.1 and 0.3 are typical.

NXY. This is the number of individual computational cells along the edge of the entire computational grid. The FFT routine is more efficient when NXY is a power of 2, and NXY is normally entered as 64.

NCW. This parameter permits CW propagation to be included by allowing the summation in Eq. (72) to be replaced by an integral [7]. Before the summation is replaced, Eq. (72) can be written in terms of physical parameters as

$$\frac{3N(\gamma - 1)k^2 \alpha E_p\, e^{-\alpha z}}{c_s^2} \sum_{n=1}^{\infty} |\Phi[x - n(v_0 + \Omega z)\Delta t, y, z]|^2 .$$

This summation is performed when NCW = 0. When NCW = 1, the program is in the CW mode, and Eq. (72) is replaced by

$$\frac{3N(\gamma - 1)k^2 \alpha P e^{-\alpha z} \sqrt{D_1}}{c_s^2 (v_0 + \Omega z)} \int_{-\infty}^{0} |\Phi(x + x', y, z)|^2\, dx',$$

where P is the average power of a CW laser ($P = E_p/\Delta t$). The integration is performed using a simple trapezoid rule.

NAD. When NAD = 0, the coordinate system adaption is not included. When NAD = 1, it is included.

NMS. When NMS = 0, the midplane integrations are not used. When NMS = 1, they are used.

NPM. When NPM = −1, the second data card contains physical parameters. When NPM = +1, the second card contains dimensionless parameters.

NBM. This parameter selects one of the five beamshapes available within the program:

NBM = 0 — Infinite Gaussian, with WIDTH (a parameter read from the second input card) being the e^{-1} intensity radius;

NBM = 1 — Truncated Gaussian, with WIDTH being the e^{-1} intensity radius, truncated at $\sqrt{2} \times$ WIDTH or e^{-2} intensity radius;

NBM = 2 — Uniform circular aperture, with WIDTH being the actual aperture radius;

NBM = 3 — Uniform square aperture, with WIDTH being the dimension from the center of the square to the edge (half-side dimension) in the x or y direction;

NBM = 4 — Uniform circular aperture and an occulting disk, with WIDTH being the total aperture radius and, as stated previously, with ROCULT being the ratio giving the occulting disk radius;

NBM = 5 — Uniform rectangular aperture, with WIDTH being the half-side x dimension and ROCULT being the ratio giving the y dimension.

NPLOT. This determines the type and the number of plots given in the output:

NPLOT = 0 — No plots;

NPLOT = 1 — Final contour plot only;

NPLOT = 2 — Final contour plot plus a plot of average intensity and peak intensity versus z;

NPLOT = 3 — Preceding plots plus a plot of flux and area versus irradiance;

NPLOT = 4 — Preceding plots plus a contour plot of aperture intensity;

NPLOT = 5 — Preceding plots plus Fourier-transform contour plots of aperture and final intensity distributions.

NCT. This determines the contour levels used in the contour plots:

NCT = 0 — Contour plots use contour levels with 10% increments;

NCT = 1 — Contour plots use 3-dB contours (0.5^n, $n = 1, 2, \ldots, 10$).

NRS. When NRS = 1, the final contour plot is corrected and standardized according to an internal criterion, to remove the effects of different amounts of coordinate system adaption in the x and y directions. When NRS = 0, this plot can appear with nonuniform axes.

NPUNCH. This determines whether there is a punched-card output:

NPUNCH = 0 — No punched-card output;

NPUNCH = 1 — Punched-card output for later data processing.

NID. Up to six characters can be used to identify a run or a series of runs on both the printed and punched output.

Second Input Card

The data contained on the second input card depend on the value of NPM. If NPM = −1, the physical parameters listed in Table 2 will be read. A description of each of these parameters is as follows:

OM. The slew rate in radians per second.

HT. The interval between pulses in seconds, or the reciprocal of the pulse repetition frequency (PRF). For CW propagation this should be set to 1 second.

ALPHA. The absorption coefficient α in km^{-1}.

ALPHAS. The scattering coefficient in km^{-1}. ALPHAS is used to compute the total extinction but is not included in the absorption that produces atmospheric heating.

WIDTH. The aperture radius a in centimeters. The particular definition is given in the preceding subsection for each value of NBM.

Table 2—Parameters Specified by the Second Input Card When NPM = −1

Columns	Name	Format
1-5	OM	F5.0
6-10	HT	F5.0
11-15	ALPHA	F5.0
16-20	ALPHAS	F5.0
21-30	WIDTH	E10.0
31-40	WN	E10.0
41-50	VO	E10.0
51-60	ENERGY	E10.0
61-70	F	E10.0
71-80	ZF	E10.0

WN. The wavenumber $k = 2\pi/\lambda$ or $2\pi/\beta\lambda$, where β is the beam quality and λ is the beam wavelength in centimeters.

VO. The wind velocity v_0 in meters per second.

ENERGY. The individual pulse energy E_p in joules. For CW propagation ENERGY is the average power in watts.

F. The focal length in kilometers.

ZF. The distance at which the calculation is to be stopped in kilometers.

As already shown, the propagation is a function of five dimensionless parameters. Different combinations of the eight physical parameters, which are required to define

UNCLASSIFIED

these dimensionless parameters and which lead to the same values of the dimensionless parameters, will produce identical results. In order that a unique physical situation be specified, some physical quantities are also read from the second data card when NPM = +1 (Table 3). They are not used to define the physical situation but rather to assign units to the derived quantities at the end of the calculations. The quantities read when NPM = +1 are:

Table 3—Quantities Specified by the Second Input Card When NPM = +1

Columns	Name	Format
1-5	F	F5.0
6-10	HT	F5.0
11-20	PNA	E10.0
21-30	PNALF	E10.0
31-40	PNK	E10.0
41-50	PNO	E10.0
51-60	PNS	E10.0
61-70	PND	E10.0
71-80	PNZ	E10.0

F. Focal length in kilometers.

HT. Pulse interval Δt in seconds (=1 second for CW).

PNA. The f number = WIDTH/F.

PNALF. Absorption number, ALPHA/F.

PNK. Fresnel number, WN $\cdot$ WIDTH^2/F.

PNO. Overlap number, $2\sqrt{2}\cdot$WIDTH/(VO$\cdot$HT) for an infinite and truncated Gaussian beam and 2$\cdot$WIDTH/(VO$\cdot$HT) for all other beam shapes.

PNS. Slew number, OM$\cdot$F/VO.

PND. Distortion number, $3Nk(\gamma - 1)\alpha f E_p / c_s^2 a\, v_0 \Delta t$.

PNZ. The ratio of the distance at which the calculation is to be stopped to the focal length, ZF/F.

Examples of Output

A series of multipulse runs was made varying the pulse spacing and energy so that the average power remained constant and using a number of average powers. The results of these runs are shown in Fig. 2 in the form of power optimization curves. The CW curve is included so that the convergence of the multipulse curves to the CW curve, as the limiting case when pulse interval is decreased, can be readily observed.

To test the SSPARAMA code in the CW mode, some comparison runs were made to check against some results obtained from Jan Herrmann of Lincoln Laboratory, who studied the propagation of a CW infinite Gaussian with a e^{-2} diameter of 70 cm. The absorption coefficient was 0.07 km^{-1}, with no scattering. The laser was twice-diffraction-limited DF with a wavenumber of 8.5×10^3 cm^{-1}. Two cases were considered at focal lengths of 2, 5, and 10 km. The first case had a power of 10 MW, a wind speed of 250 m/s, and no slewing. The second case had 2 MW power, a 2-m/s wind, and a 0.02-s^{-1} slew. The results, consisting of the area containing 63% of the focal-plane power and of the peak intensity are summarized in Table 4. A_{rel} and I_{rel} compare these quantities with those that would have been obtained if there were no thermal blooming. The results for these highly bloomed cases agree within about 5% with those of Herrmann.

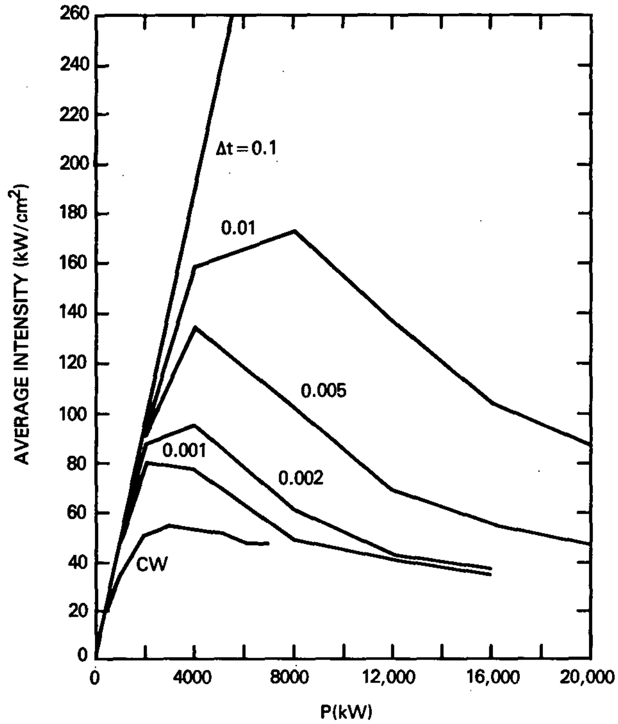

Fig. 2—SSPARAMA results (F = 1 km, diam = 70.7 cm (1/e), α = 0.1 km^{-1}, k = 2966 cm^{-1}, v_0 = 10 m/s, and Ω = 0.1)

Table 4—SSPARAMA Results for the Propagation of a CW Infinite Gaussian With a Wavenumber of 8500 cm^{-1}, an e^{-2} Diameter of 70 cm, an Absorption Coefficient of 0.07 km^{-1}, and No Scattering

Focal Length F (km)	Area A Containing 63% of the Focal-Plane Power (cm^2)	Relative Area A_{rel} Relative To No Thermal Blooming	Peak Intensity I_{peak} (kW/cm^2)	Relative Peak Intensity I_{rel} Relative To No Thermal Blooming
First Case: 10 MW Power, 250-m/s Wind, and No Slew				
2	57.6	20.3	147	0.0464
5	658	37.0	10.3	0.0251
10	3543	49.8	1.33	0.0184
Second Case: 2 MW Power, 2-m/s Wind, and 0.02-s^{-1} Slew				
2	64.8	22.8	26.8	0.0422
5	474	26.6	2.96	0.0359
10	2018	28.4	0.495	0.0341

Another example of SSPARAMA output is illustrated in Fig. 3, namely, the final contour plot for the 5-km run from the first case with 10% contour levels. The complete printed output from SSPARAMA is included in Figs. 4a through 4c.

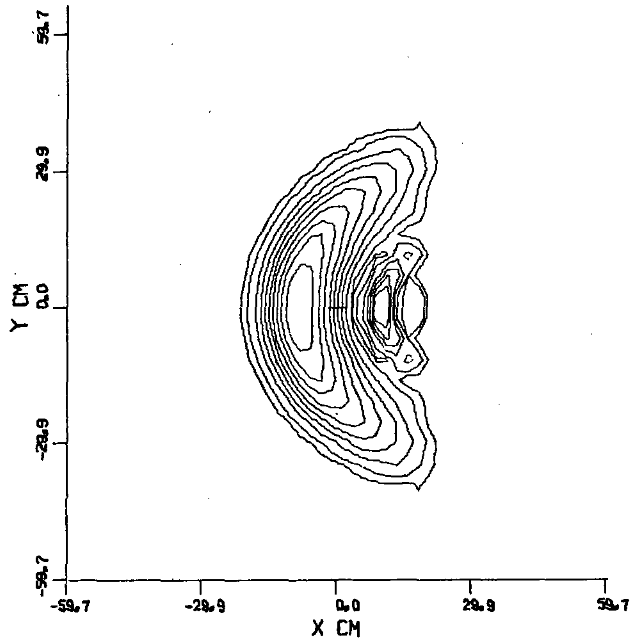

Fig. 3—Contour plot with 10% contour levels for the 5-km run from the first case in Table 4 (PNALF = 0.350, PNK = 10.400, PNO = 0.002, PNS = 0.000, PND = 80.000)

Figure 4a, the first page of printed output, is almost self-explanatory. Both dimensionless and physical parameters are listed; one is computed from the other, depending on which was entered. The program options indicate the mode, either CW or MP and the beamshape etc. The results summary in Fig. 4a includes the final value of the energy conservation integral, Eq. (2). This quantity, which is ideally equal to 1, gives a quick check on the validity of the numerical calculations. One factor that limits the accuracy is the use of a finite mesh size. As this mesh is made finer, the intensity distribution gets closer to the mesh boundaries, and numerical errors may enter through diffraction and the use of a discrete Fourier-transform routine as energy is reflected off the boundary. To avoid this reflection, the outermost boundary of the computational grid is set to zero and the next outermost boundary is set to one half its value at each z step. Thus the sum over normalized intensity gives an indication of how much energy was lost due to boundary-value problems.

The area that is given in Fig. 4a is the area containing exactly 0.63 of the total flux obtained by linear interpolation between adjacent flux fractional areas. This area will include contributions from several peaks as the intensity pattern breaks up under severe blooming conditions, so its meaning may also require a suitable interpretation of the intensity contour map. In addition the relative area and maximum intensity are calculated relative to the focal area and intensity of a vacuum-propagated infinite Gaussian whose e^{-1} diameter is equal to the value of WIDTH regardless of the beamshape being propagated.

```
                         *** MEPHISTO INPUT DATA ***

DIMENSIONLESS PARAMETERS      PHYSICAL PARAMETERS          NUMERICAL PARAMETERS

    NA =  0.000049              RADIUS(M) =      0.248          HX = 0.20
  NALF =     0.350          ALPHA(1/KM) =   0.070000          HY = 0.20
    NK =     10.40              K(1/CM) =    8488.93          NX =   64
    NG =      0.00                V(M/S) =    249.98          NY =   64
    NS =      0.00        OMEGA(RAD/SEC) =     0.0000      PHIMXX =1.000
    NC =     80.00            ENERGY(KJ) =   10420.69
    NZ =    1.0000                 F(KM) =      5.000
                                 DT(SEC) =   1.000000
                            ALPHAS(1/KM) =   0.000000


PROGRAM OPTIONS

                       MODE      CW
                  BEAMSHAPE      INFINITE GAUSSIAN
                   ADAPTION      YES
     HALF-STEP INTEGRATION      YES
        PUNCHED CARD OUTPUT      NO
            NUMBER OF PLOTS        5
         LOW LEVEL CONTOURS      NO
   RESCALE FINAL CONTOUR PLOT    YES


                                        *** RESULTS ***

                        THE CALCULATIONS REACHED Z =   5.00000 (KM)

                  THE SUM OVER NORMALIZED INTENSITY =   1.00000

                               THE NUMBER OF Z-STEPS =   22

        AVERAGE POWER (KW) EMITTED AT APERTURE = 10420.694

                 AVERAGE TRANSMITTED POWER (KW) =  7343.339

          AREA (SQCM) CONTAINING 0.63 OF POWER =   657.995

              A REL (RELATIVE TO INF. GAUSSIAN) =    36.982

     AVERAGE INTENSITY (KW/SQCM) IN THIS AREA =     7.031

                      PEAK INTENSITY (KW/SQCM) =    10.345

         I REL (RELATIVE TO INF. GAUSSIAN PEAK) =   0.02507
```

Fig. 4a—First page of the output by SSPARAMA, containing the input that resulted in Fig. 3 and a summary of the results

Figure 4b, the page containing numerical data, begins with a list of internally computed quantities that relate to the problems of air breakdown and t-cubed self-blooming. They are printed only for possible future data analysis. Assuming the breakdown intensity at 10.6 μm is 3×10^6 W/cm^2 and that this is inversely proportional to wavelength squared, the following quantities are computed as a function of range: the minimum area required for breakdown, the ratio of this minimum area to the vacuum area, the maximum pulselength before breakdown occurs, the critical power, the saturation time, the intensity produced by the critical power propagating in a vacuum, and factors accounting for turbulence with values of C_n^2 of 10^{-15} and 10^{-14}. This is followed by an x and y slice through the aperture to check the initial beamshape.

The quantities, including the values of HZN in z/ka^2 units, relating to the coordinate system adaption are printed at each z step. The headings D, D1, D2, ALPHA1, ALPHA2, BETA1, DALPH1, DALPH2, DBET1, and XCEN correspond to D, D_1, D_2, α_1, α_2, β, $\Delta\alpha_1$, $\Delta\alpha_2$, $\Delta\beta$, and X used in the second section of this report. Also included is EPSMX, the maximum value of the summation given in Eq. (72); PHIMX, the maximum value of the positive phase change applied to ψ to obtain Φ; and PARM, the number of pulses, for the MP mode, that occur in a computational cell.

Figure 4c, the output data, lists in the top portion the area, flux, the area fraction, and flux fraction contained within each contour level. From these data the 63% area is interpolated. This is followed in the middle portion by the z locations of the maximum of the average and peak intensities, the minimum 63% area, and the minimum z step that

*** NUMERICAL DATA ***

RANGE(KM)	AS2(CM2)	AS2/A2D	TP(SEC)	PCR(W)	ISAT(SEC)	ISAT(W/CM2)	TURBCOR1	TURBCOR2
0.05000	98.99318	0.16193	5.4439-003	1.0429+005	5.4483-005	4.0712+001	1.0000+000	1.0000+000
0.56056	1018.03493	2.10781	5.0917-004	5.1276+005	2.5054-005	2.4370+002	9.9997-001	9.9947-001
1.11611	1395.21180	3.77194	3.5735-004	6.5774+005	2.2555-005	3.9261+002	9.9961-001	9.9378-001
1.67167	1658.41931	6.09563	2.8917-004	7.6732+005	2.1293-005	5.9895+002	9.9805-001	9.6992-001
2.22722	1859.23530	9.81094	2.4810-004	8.6091+005	2.0497-005	9.2798+002	9.9301-001	8.9967-001
2.78278	2018.48219	16.51645	2.1981-004	9.4546+005	1.9943-005	1.5200+003	9.7823-001	7.3929-001
3.33833	2147.25482	30.59665	1.9875-004	1.0243+006	1.9536-005	2.7584+003	9.3513-001	4.7630-001
3.89389	2252.39000	67.41011	1.8224-004	1.0994+006	1.9227-005	5.9811+003	8.0745-001	2.0923-001
4.44944	2338.50222	196.31625	1.8884-004	1.1720+006	1.8988-005	1.7202+004	4.9384-001	5.7989-002
5.00500	2408.90687	424.44483	1.9765-004	1.2428+006	1.8803-005	3.6826+004	2.4185-001	1.9731-002

NORMALIZED AMPLITUDE SAMPLES AT APERTURE

AT X= 32 Y= 1 TO 64

0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.1 0.1 0.1 0.2 0.2 0.3 0.4 0.4 0.5 0.5 0.6
0.6 0.5 0.5 0.4 0.4 0.3 0.2 0.2 0.1 0.1 0.1 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

AT Y= 32 X= 1 TO 64

0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.1 0.1 0.1 0.2 0.2 0.3 0.4 0.4 0.5 0.5 0.6
0.6 0.5 0.5 0.4 0.4 0.3 0.2 0.2 0.1 0.1 0.1 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

Z	HZ	D	D1	D2	ALPHA1	ALPHA2	BETA1	DALPH1	DALPH2	DBET1	XCEN	EPSMX	PHIMX	PARM
0.224	2.2-003	0.9124	0.9124	0.9124	-4.9649	-4.9649	0.0000	0.0001	0.1749	-0.2820	0.0000	5.5-004	0.0000	2.0-004
0.442	4.2-003	0.8311	0.8311	0.8339	-4.7360	-4.5769	-0.2692	0.0009	0.1781	-0.2833	-0.0306	5.2-004	0.4949	1.9-004
0.655	4.1-003	0.7555	0.7555	0.7685	-4.5120	-4.2066	-0.5267	0.0024	0.1808	-0.2839	-0.0920	5.0-004	0.4935	1.8-004
0.863	4.0-003	0.6849	0.6849	0.7004	-4.2912	-3.8530	-0.7719	0.0049	0.1857	-0.2851	-0.1845	4.7-004	0.4919	1.7-004
1.068	4.0-003	0.6189	0.6190	0.6438	-4.0719	-3.5131	-1.0048	0.0086	0.1934	-0.2870	-0.3091	4.4-004	0.4903	1.6-004
1.271	3.9-003	0.5570	0.5574	0.5931	-3.8525	-3.1833	-1.2258	0.0136	0.2043	-0.2898	-0.4673	4.1-004	0.4885	1.6-004
1.471	3.9-003	0.4988	0.4996	0.5480	-3.6310	-2.8601	-1.4350	0.0204	0.2194	-0.2937	-0.6615	3.9-004	0.4867	1.5-004
1.672	3.9-003	0.4441	0.4456	0.5081	-3.4058	-2.5392	-1.6325	0.0295	0.2396	-0.2990	-0.8946	3.6-004	0.4850	1.4-004
1.872	3.9-003	0.3927	0.3951	0.4733	-3.1748	-2.2156	-1.8189	0.0414	0.2664	-0.3062	-1.1706	3.3-004	0.4832	1.3-004
2.073	3.9-003	0.3442	0.3461	0.4436	-2.9360	-1.8830	-1.9947	0.0572	0.3017	-0.3161	-1.4946	3.0-004	0.4815	1.2-004
2.277	3.9-003	0.2986	0.3046	0.4193	-2.6871	-1.5320	-2.1615	0.0780	0.3478	-0.3301	-1.8735	2.8-004	0.4799	1.2-004
2.484	4.0-003	0.2554	0.2644	0.4006	-2.4240	-1.1335	-2.3214	0.1067	0.4101	-0.3538	-2.3199	2.5-004	0.4744	1.1-004
2.696	4.0-003	0.2150	0.2281	0.3886	-2.1443	-0.7271	-2.4849	0.1445	0.4853	-0.3864	-2.8435	2.3-004	0.4772	1.0-004
2.912	4.1-003	0.1775	0.1961	0.3847	-1.8449	-0.2352	-2.6624	0.1945	0.5707	-0.4370	-3.4622	2.1-004	0.4764	9.5-005
3.134	4.2-003	0.1429	0.1690	0.3905	-1.5209	0.3535	-2.8774	0.2579	0.6491	-0.5090	-4.2020	1.9-004	0.4761	8.8-005
3.367	4.4-003	0.1109	0.1473	0.4084	-1.1654	1.0209	-3.1616	0.3250	0.6899	-0.5920	-5.1128	1.7-004	0.4705	8.1-005
3.610	4.6-003	0.0821	0.1321	0.4412	-0.7796	1.7937	-3.5548	0.3473	0.6543	-0.6258	-6.2478	1.6-004	0.4774	7.6-005
3.869	4.8-003	0.0567	0.1239	0.4911	-0.3994	2.5777	-4.0476	0.3004	0.5712	-0.5733	-7.6632	1.5-004	0.4793	7.2-005
4.142	5.1-003	0.0358	0.1220	0.5595	-0.0769	3.3661	-4.5864	0.2350	0.4881	-0.4779	-9.3684	1.5-004	0.4819	7.0-005
4.431	5.4-003	0.0202	0.1258	0.6484	0.1831	4.1340	-5.1423	0.1912	0.4283	-0.4030	-11.3604	1.5-004	0.4852	6.9-005
4.740	5.7-003	0.0110	0.1352	0.7622	0.4112	4.9725	-5.7477	0.1636	0.3889	-0.3554	-13.6639	1.6-004	0.4884	7.0-005
5.000	6.2-003	0.0092	0.1471	0.8731	0.6177	5.7417	-6.3666	0.1158	0.2889	-0.2570	-15.7174	1.7-004	0.4909	7.3-005

Fig. 4b—Second page of the output, containing numerical data

have occurred during beam propagation. Then in the lower portion the peak and average intensities, the 63% area and the location of the peak intensity in centimeters are listed at each z step.

Summary of Program Structure

When the half-step integrations are used, the solution is advanced twice before the information at each z step is stored. This can be seen from the flow chart of SSPARAMA (Fig. 5). Thus, when NMS = 1, the program actually used twice the number of z steps that are printed and included points approximately midway between those listed.

The structure of the code SSPARAMA is explained below and summarized in the flow chart in Fig. 5.

- The call to subroutine START causes the input data to be read. The real part of the 64-by-64 array ψ is defined according to the beamshape specified. Initially the phase of this array is zero.

*** OUTPUT DATA ***

LEVEL	AREA (SQ CM)	FLUX (KW)	AREA FRACTION	FLUX FRACTION	IRRADIANCE (KW/SQ-CM)
0.9000	8.781+001	8.615+002	0.0556	0.1173	9.811+000
0.8000	1.756+002	1.630+003	0.1111	0.2219	9.280+000
0.7000	2.634+002	2.320+003	0.1667	0.3159	8.806+000
0.6000	4.039+002	3.274+003	0.2556	0.4458	8.105+000
0.5000	5.795+002	4.261+003	0.3667	0.5803	7.353+000
0.4000	7.200+002	4.915+003	0.4556	0.6693	6.826+000
0.3000	9.483+002	5.754+003	0.6000	0.7836	6.067+000
0.2000	1.177+003	6.323+003	0.7444	0.8611	5.374+000
0.1000	1.581+003	6.930+003	1.0000	0.9438	4.385+000
0.0500	1.826+003	7.126+003	1.1556	0.9704	3.902+000

MAXIMUM AVG I = 1.873+001 AT Z= 3.367+000
MAXIMUM PEAK I= 3.841+001 AT Z= 3.610+000
MINIMUM AREA = 2.741+002 AT Z= 3.610+000
MINIMUM HZ = 2.170-003 AT Z= 1.000-006

Z	IAVE	A63	IMAX	XPEAK	YPEAK
0.000	3.421	1918.802	5.308	0.000	0.000
0.224	3.688	1752.298	5.719	0.000	0.000
0.442	3.973	1602.249	6.152	0.000	0.000
0.655	4.277	1466.170	6.608	0.000	0.000
0.863	4.604	1342.439	7.083	0.000	4.143
1.068	4.970	1225.665	7.570	0.000	3.972
1.271	5.361	1120.431	8.060	0.000	3.812
1.471	5.792	1022.527	8.533	0.000	3.664
1.672	6.274	930.908	9.217	-3.304	3.528
1.872	6.791	848.033	9.930	-3.112	0.000
2.073	7.430	764.166	10.776	-5.841	3.297
2.277	8.260	677.725	12.064	-5.464	3.205
2.484	9.320	591.981	13.519	-7.636	3.133
2.696	10.786	503.964	16.524	-7.092	3.086
2.912	12.922	414.367	20.387	-6.576	6.140
3.134	15.727	335.210	25.611	-6.105	6.186
3.367	18.727	276.968	32.220	-5.700	6.327
3.610	18.601	274.121	38.413	-5.398	6.576
3.869	15.599	321.008	33.563	-5.227	3.469
4.142	12.970	378.764	21.360	-3.458	3.703
4.431	10.239	470.186	16.150	-5.267	0.000
4.740	8.277	569.200	12.452	-5.460	0.000
5.000	7.031	657.995	10.345	-5.695	0.000

Fig. 4c—Third page of the output, containing the remaining numerical data

- The initialization procedure continues with the call to INTENS, where the aperture intensity is computed at each mesh point.

- The call to DENS computes the quantity $g(x, y, z)$ given in Eq. (63) and then applies the phase change given by Eq. (62) which converts ψ to Φ. The first z increment is also computed.

- The main program loop begins here with a call to OUTPUT to store various values until the calculations are completed.

- The call to ADVANCE applies the Fourier transform of Eq. (67) and then the phase change of Eq. (68). The array is Fourier-transformed back to yield $\Phi(z + \Delta z)$.

- The intensity is computed with the call to INTENS, and the boundary values of the array are tapered to zero.

- The call to DENS now includes a call to VTRANS, by which the phase change of Eq. (62) is reversed, converting Φ back to ψ. The quantities $\{\alpha_1, \alpha_2, \beta\}$ and $\{\Delta\alpha_1, \Delta\alpha_2, \Delta\beta\}$ are found in VTRANS, and the values of D_1 and D_2 are updated. After the return to DENS, Eq. (63) is solved and the phase change of Eq. (62) is reapplied, converting ψ back to Φ in preparation for the next call to ADVANCE.

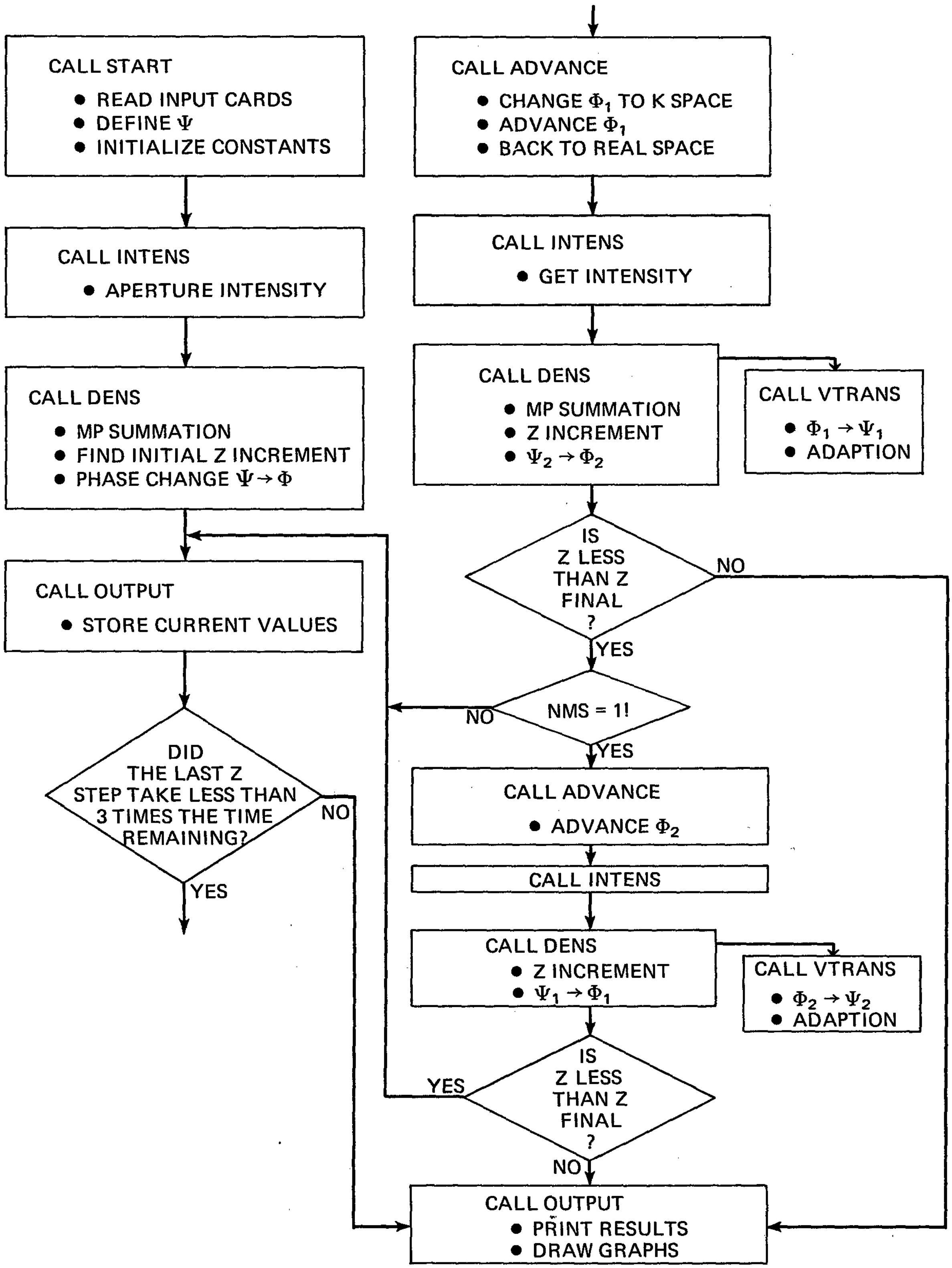

Fig. 5—Summary of the code SSPARAMA

- Now that one cycle of propagating the solution is completed, the code checks if z final has been reached and if the half-step integrations are to be performed as outlined in the section titled Numerical Procedures.

- When z final has been reached or the time limit of execution is near, the last call to OUTPUT prints the results and ends this run.

The Appendix contains a complete listing of the code with copious comments included.

REFERENCES

1. P.B. Ulrich, J.N. Hayes, J.H. Hancock, and J.T. Ulrich, "Documentation of PROP E, a Computer Program for the Propagation of High Power Laser Beams Through Absorbing Media," NRL Report 7681, May 29, 1974.
2. P.B. Ulrich, "PROP-I: An Efficient Implicit Algorithm for Calculating Nonlinear Scalar Wave Propagation in the Fresnel Approximation," NRL Report 7706, May 29, 1974.
3. A.H. Aitken, J.N. Hayes, and P.B. Ulrich, "Propagation of High-Energy 10.6-Micron Laser Beams Through the Atmosphere," NRL Report 7293, May 28, 1971.
4. J. Wallace and J.Q. Lilly, "Thermal blooming of repetitively pulsed laser beams," *J. Opt. Soc. Am.* 64, 1651–1655 (Dec. 1974).
5. J. Herrmann and L.C. Bradley, "Numerical Calculation of Light Propagation," Massachusetts Institute of Technology, Lincoln Laboratories, Lexington, Mass. Report LTP-10, July 12, 1971.
6. H.J. Breaux, "An Analysis of Mathematical Transformations and a Comparison of Numerical Techniques for Computation of High Energy CW Laser Propagation in an Inhomogeneous Medium," BRL Report 1723, June 1974.
7. K.G. Whitney and P.B. Ulrich, "Scaling Laws for Multipulse Steady-State Thermal Blooming," NRL Memorandum Report 3229, Mar. 1976, Appendix B3.

APPENDIX A
Listing of Code and Comments

```
      PROGRAM SSPARAMA
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      COMMON /999/ A1(64,64), A2(64,64), ITENS(64,64)
     COMMON /AAA/ EPS(64,64),EPSO(64,64),AOUT(11,99),BOUT(9,10),
    . AIN(2,64),DALPH1(2),DALPH2(2),DBET1(2),ALPH10(2),ALPH20(2),
    .   BET10(2), D10(2), D20(2), RD10(2), RD20(2), SRTD10(2), XCENO(2)
      COMMON /BBB/ TENS(64,64), G1(64), G2(64), PHASE1(64), PHASE2(64),
    . CONMIN(10),M2(3),SV1(64),SV2(64),PARM(80)
      COMMON /SINGLS/ F, PNA, PNALF, PNK, PNO, PNS, PND, PNZ, HX, HY,
    .   HZ, Z, ZZ,ZZF, ZNM, ZFINAL, XZERO, YZERO, WIDTH, ALPHA, WN,
    .   VODT, OMDT, HT, ENERGY, ALPHAC, CS, REFRAC, GAMMA, ETJ, CTK,
    .  EJTKJ, RHT, POUT, DAREA, W2, TS, TPULSE, AS2,PCR,SI,TCOR1,TCOR2,
    .   Z1, RI63MX, Z2, RIMXMX, Z3, APMN, Z4, HZMN, DKAREA, TENSMX,
    .   EX, PHIMX, EPSMX, ERRMX, DGMX, R1, BDIMAX, VTERM, PHIMXX,HZNMS,        14
    .   PI, IMAX, JMAX, NX, NY, NAD, NX2, NY2, NXY, NXDIM, NYDIM, NPT,
    .   IPLOT,NITER,NBUF,NXM,NYM,NMS,NFLAG,D,D1,D2,P1,P2,SRTD1,SRTD2,
    .   RSRD12,XCEN,TLAST,SQRT8,PNDO,GCONO,GCON,BDI,HCZ10,HCZ20,HCZ1N,
    .   HCZ2N,HCZ12,ALPH1,ALPH2,BET1,CON1,CON2,HZO,HZN,EXO,EXN,WT1,WT2
     COMMON /OUTS/ NBM,SCLFAC,NRS,NPM,NCW,NEXIT,NPLOT,NPUNCH
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      COMPLEX A1, A2
      LOGICAL LS
      DATA (CS=34000.0), (REFRAC=0.154), (GAMMA=1.4), (ETJ=1.0E-7),
    .   (CTK=1.0E-5),  (PI=3.14159265), (BDI=3.0E6), (EJTKJ=1.0E-3)
      DATA (NXDIM=64), (NYDIM=64), (NZ=20)
      BANK,(0),/999/
C
C     INPUT AND INITIALIZATION
C
      TSTART=TIMELEFT(DUMMY)
      LS=.FALSE.
55    CONTINUE
      CALL START(LS)
      NEXIT=0
      NITER=0
      IPLOT=0
      ZZ1=0.0
      ZZ2=0.0
      I2=2
      IF (NMS .EQ. 0) I2=1
C
      CALL INTENS(A1,.FALSE.)
      CALL DENS(A1, A1, ZZ1, 1, 1, .FALSE.)
      HZO=0.0
C
C*************************************************************************
C
C    MAIN PROGRAM LOOP
C
14     CONTINUE                                                         C
       NITER=NITER+1
C
C    STORE VALUES FOR LATER PRINTOUT
C
2      CALL OUTPUT(.FALSE.)
C
C    IF TIME REMAINING IS LESS THAN 3 TIMES THAT FOR THE LAST
C    Z STEP - EXIT
C
```

```
   3    TNOW=TIMELEFT(0)
        DT=TLAST-TNOW
        TLAST=TNOW
        IF(3*DT.LE.TNOW) GO TO 8
        PRINT 22,Z
  22    FORMAT(//25X25H*** TIME ABORT AT Z(KM) =F10.5,1X3H***//)
        GO TO 13
    8   CONTINUE
C
C       ADVANCE FROM ZZ TO ZZ+DZ, CALCULATING NEW AMPLITUDES IN A
C
  40     CALL ADVNCE(A1,1)
         CALL INTENS(A1, .FALSE.)
         CALL DENS(A1, A2, ZZ1, 1, I2, .TRUE.)
         IF (Z .GE. ZFINAL) GO TO 15
C                                                                   C
C     REPEAT IF HALF-STEP INTEGRATION IS INCLUDED
C
         IF (NMS .EQ. 0) GO TO 45
         CALL ADVNCE(A2,2)
         CALL INTENS(A2, .FALSE.)
         CALL DENS(A2, A1, ZZ2, 2, 1, .TRUE.)
         IF (Z .GE. ZFINAL) GO TO 15
  45    CONTINUE
         GO TO 14
C
C*******************************************************************************
C
C     SET NEXIT EQUAL 1 FOR PREMATURE EXITS
C
  13    NEXIT=1
  15    CONTINUE
C
C     EXECUTE ALL OUTPUT
C
        CALL OUTPUT(.TRUE.)
        PRINT 16
  16    FORMAT(1H1)
  17    CALL STOPPLOT

C           PRINT RUN TIME (CP TIME).
        TRUN=(TSTART-TIMELEFT(DUMMY))/60.
        PRINT 18,TRUN
   18 FORMAT(//,16(1H*),*  RUN TIME=*,F6.2,* MINUTES*)

        STOP
        END
```

```
      SUBROUTINE START(LS)
C
C    THIS SUBROUTINE READS THE INPUT PARAMETERS AND DEFINES
C    THE APERTURE DISTRIBUTION PSI (OR A).  MANY QUANTITIES ARE
C    INITIALIZED HERE FOR LATER USE.
C
C***********************************************************************
C
      COMMON /999/ A1(64,64), A2(64,64), ITENS(64,64)
      COMMON /AAA/ EPS(64,64),EPSO(64,64),AOUT(11,99),BOUT(9,10),
     . AIN(2,64),DALPH1(2),DALPH2(2),DBET1(2),ALPH10(2),ALPH20(2),
     .   BET10(2), D10(2), D20(2), RD10(2), RD20(2), SRTD10(2), XCENO(2)
      COMMON /BBB/ TENS(64,64), G1(64), G2(64), PHASE1(64), PHASE2(64),
     . CONMIN(10),M2(3),SV1(64),SV2(64),PARM(80)
      COMMON /SINGLS/ F, PNA, PNALF, PNK, PNO, PNS, PND, PNZ, HX, HY,
     .   HZ, Z, ZZ,ZZF, ZNM, ZFINAL, XZERO, YZERO, WIDTH, ALPHA, WN,
     .   VODT, OMDT, HT, ENERGY, ALPHAC, CS, REFRAC, GAMMA, ETJ, CTK,
     .  EJTKJ, RHT, POUT, DAREA, W2, TS, TPULSE, AS2,PCR,SI,TCOR1,TCOR2,
     .   Z1, RI63MX, Z2, RIMXMX, Z3, APMN, Z4, HZMN, DKAREA, TENSMX,
     .   EX, PHIMX, EPSMX, ERRMX, DGMX, R1, BDIMAX, VTERM, PHIMXX,HZNMS,      14
     .   PI, IMAX, JMAX, NX, NY, NAD, NX2, NY2, NXY, NXDIM, NYDIM, NPT,
     .   IPLOT,NITER,NBUF,NXM,NYM,NMS,NFLAG,D,D1,D2,P1,P2,SRTD1,SRTD2,
     .   RSRD12,XCEN,TLAST,SQRT8,PNDO,GCONO,GCON,BDI,HCZ10,HCZ20,HCZ1N,
     .   HCZ2N,HCZ12,ALPH1,ALPH2,BET1,CON1,CON2,HZO,HZN,EXO,EXN,WT1,WT2
      COMMON /OUTS/ NBM,SCLFAC,NRS,NPK,NCW,NEXIT,NPLOT,NPUNCH
      COMMON/PUN/NID
C***********************************************************************
C
      COMPLEX A1, A2
      LOGICAL LS, LP, LT
      DIMENSION IPWR(64)
      DIMENSION PBUF(512), PV(64)
C
C     INPUT AND INITIALIZATION
C
      DTURBF(DUMY)=(1.0-ZNM)**2+((ZNM/PNK)**2)*(1.0+1.33333333*WIDTH*
     .   WIDTH*CTK*((PI*PI*DUMY*WN*WN*ZNM*F*400.0)**(6./5.)))
      CALL PLOTS (PBUF,512,1)
      SCLFAC=10.0
      IF (LS) GO TO 30
C
C***********************************************************************
C
C    READ INPUT DATA - PROGRAM OPTIONS
C
C    NCW = 0 - MULTIPULSE MODE
C    NCW = 1 - CW MODE
C    NAD = 0 - NO ADAPTION
C    NMS = 0 - NO HALF STEP INTEGRATIONS
C    NPM GT 0 - READ IN DIMENSIONLESS PARAMETERS
C    NPM LT 0 - READ IN PHYSICAL PARAMETERS
C    NBM = 0 - INFINITE GAUSSIAN
C    NBM = 1 - TRUNCATED GAUSSIAN
C    NBM = 2 - UNIFORM CIRCLE
C    NBM = 3 - UNIFORM SQUARE
C    NBM = 4 - UNIFORM CIRCLE WITH ROCULT OCCULTED RADIUS
C    NPLOT = 0 - NO PLOTS
C    NPLOT = 1 - FINAL CONTOUR PLOT ONLY
C    NPLOT = 2 - ABOVE + I VS Z PLOT
C    NPLOT = 3 - ABOVE + FLUX AND AREA VS IRRADIANCE
C    NPLOT = 4 - ABOVE + APERTURE CONTOUR PLOT
C    NPLOT = 5 - ABOVE + FOURIER TRANSFORM CONTOUR PLOT OF APERTURE
C                AND FOCUS
C    NCT = 0 - 10 PERCENT CONTOUR LEVELS
C    NCT NE 0 - LOW CONTOUR LEVEL OPTION
C    NRS = 0 - DO NOT RESCALE FINAL CONTOUR PLOT
```

```
C     NPUNCH = 0 - NO PUNCHED OUTPUT
C     NID IS AN ID NUMBER READ IN A6 FORMAT
C
C     FOR GAUSSIANS - WIDTH = 1/E INTENSITY RADIUS
C     FOR UNIFORM APERTURES - WIDTH = RADIUS OR HALF SIDE DIMENSION
C
        READ 1, PHIMXX,ROCULT,HXY,NXY,NCW,NAD,NMS,NPM,NBM,NPLOT,
       1  NCT,NRS,NPUNCH,NID
      1 FORMAT(3F5.0,10I5,9X,A6)
        HX=HXY
        HY=HXY
        NX=NXY
        NY=NXY
        IF (NX.GT.NXDIM) NX=NXDIM
        IF (NY.GT.NYDIM) NY=NYDIM
C
        IF (NPM.LT.0) GO TO 23
C
C     READ IN DIMENSIONLESS PARAMETERS
C
C       F=FOCAL LENGTH IN KM
C       HT=PULSE INTEVAL IN SEC ( =1 SEC FOR CW)
C       PNA=WIDTH (1/E INTENSITY RADIUS)/F
C       PNALF = ALPHA/F
C       PNK = K * A**2 / F
C       PNO = 2 * A / (V DT)   ( DT = 1 SEC FOR CW )
C       PNS = OMEGA * F / V
C       PND = E * K * ALPHA 3N(GAMMA-1)/(V * DT * A * CS**2)
C       PNZ = Z FINAL / F (FOR DEFOCUSED CASES)
C
         READ 20, F,HT, PNA, PNALF, PNK, PNO, PNS, PND, PNZ
20      FORMAT(2F5.0,7E10.0)
        F=F*1.E+5
         WIDTH=PNA*F
         ALPHA=PNALF/F
        ALPHAS=0.0
         WN=PNK/(F*PNA*PNA)
         VODT=2.0*WIDTH/PNO
         OMDT=2.0*PNS*WIDTH/(F*PNO)
        VO=VODT/HT
        OM=OMDT/HT
         ENFFLD=CS*CS*WIDTH/(REFRAC*3.0*(GAMMA-1.0))
         PNDO=2.0*PND*PNA/(PNO*PNK*PNALF)
         GCONJ=PNDO*PNALF*PNK*PNK/PNA
         ENERGY=PNDO*ENFFLD*WIDTH*WIDTH*ETJ
         ZF=F*PNZ
        GO TO 26
C
23       IF (NPM.GT.0) GO TO 26
C
C     READ IN PHYSICAL PARAMETERS
C
C       OM= OMEGA IN RADIANS/SEC
C       HT = PULSE INTERVAL IN SECONDS
C       ALPHA= ABSORPTION IN 1/KM
C       ALPHAS= SCATTERING IN 1/KM
C       WIDTH = 1/E INTENSITY RADIUS IN CM
C       WN = 2 * PI / LAMBDA IN 1/CM
C       VO= WIND SPEED IN  M/SEC
C       ENERGY = ENERGY IN JOULES
C       F=FOCAL LENGTH IN KM
C       ZF= Z FINAL IN KM
C
        READ 24,OM,HT,ALPHA,ALPHAS,WIDTH,WN,VO,ENERGY,F,ZF
24      FORMAT(4F5.0,6E10.0)
        VO=VO*100.
```

```
      F=F*1.E+5
      ZF=ZF*1.E+5
      ALPHA=ALPHA*1.E-5
      ALPHAS=ALPHAS*1.E-5
      V0DT=V0*HT
      OMDT=OM*HT
       PNA=WIDTH/F
      PNALF=(ALPHA+ALPHAS)*F
       PNK=WN*WIDTH**2/F
       PNO=2.0*WIDTH/V0DT
       PNS=OMDT*F/V0DT
       ENFFLD=CS*CS*WIDTH/(REFRAC*3.0*(GAMMA-1.0))
       PND0=ENERGY/(ENFFLD*WIDTH**2*ETJ)
      GCON0=3.*REFRAC*(GAMMA-1.)*WN**2*ENERGY*ALPHA*1.E+7/(CS*CS)
      PND=3.*REFRAC*WN*(GAMMA-1)*ALPHA*F*ENERGY*1.E+7/
     1(CS*CS*WIDTH*V0DT)
       PNZ=ZF/F
   26 IF(NCW.EQ.1) HT=1.
C
C*****************************************************************************
C
C    PRINT INPUT PARAMETERS
C
30     CONTINUE
       PE=ENERGY/1000.
       PV0=V0/100.
       PALFS=ALPHAS*1.E5
       PALF=ALPHA*1.E5
       PW=WIDTH/100.
       PF=F/1.E5
C
       PRINT 43,NID
   43 FORMAT(70X,A6,//)
       PRINT 44,
44     FORMAT(30X27H*** MEPHISTO INPUT DATA ***///
     15X*DIMENSIONLESS PARAMETERS*,6X,*PHYSICAL PARAMETERS*
     25X*NUMERICAL PARAMETERS*/)
      PRINT 100, PNA,PW,HX,
     1PNALF,PALF,HY,
     2PNK,WN,NX,
     3PNO,PV0,NY,
     4PNS,OM,
     5PND,PE,
     6PNZ,PF,
     7HT,
     8PALFS
100    FORMAT(10X4HNA =F10.6,10X11HRADIUS(M) =F10.3,10X4HHX =F5.2,/
     18X6HNALF =F10.3, 8X13HALPHA(1/KM) =F10.6,10X4HHY =F5.2,/
     210X4HNK =F10.2,12X9HK(1/CM) =F10.2,10X4HNX =I5,/
     310X4HNO =F10.2,13X8HV(M/S) =F10.2,10X4HNY =I5,/
     410X4HNS =F10.2, 5X16HOMEGA(RAD/SEC) =F10.4/
     510X4HND =F10.2, 9X12HENERGY(KJ) =F10.2,/
     610X4HNZ =F10.4,14X7HF(KM) =F10.3,/
     736X9HDT(SEC) =F10.6,/
     831X14HALPHAS(1/KM) =F10.6,//)
       PRINT 147
147    FORMAT(5X*PROGRAM OPTIONS*/)
       IF (NCW.EQ.0) PRINT 166
       IF (NCW.EQ.1) PRINT 167
166    FORMAT(25X4HMODE5X2HMP)
167    FORMAT(25X4HMODE5X2HCW)
       IF (NBM.EQ.0) PRINT 148
       IF (NBM.EQ.1) PRINT 149
       IF (NBM.EQ.2) PRINT 150
       IF (NBM.EQ.3) PRINT 151
       IF (NBM.EQ.4) PRINT 168,ROCULT
```

```
      IF (NAD.EQ.0) PRINT 152
      IF (NAD.EQ.1) PRINT 153
      IF (NMS.EQ.0) PRINT 154
      IF (NMS.EQ.1) PRINT 155
      IF (NPUNCH.EQ.0) PRINT 156
      IF (NPUNCH.EQ.1) PRINT 157
      PRINT 158, NPLOT
      IF (NCT.EQ.0) PRINT 162
      IF (NCT.NE.0) PRINT 163
      IF (NRS.EQ.0) PRINT 164
      IF (NRS.NE.0) PRINT 165
148   FORMAT(20X9HBEAMSHAPE5X*INFINITE GAUSSIAN*)
149   FORMAT(20X9HBEAMSHAPE5X*TRUNCATED GAUSSIAN*)
150   FORMAT(20X9HBEAMSHAPE5X*UNIFORM CIRCLE*)
151   FORMAT(20X9HBEAMSHAPE5X*UNIFORM SQUARE*)
152   FORMAT(21X8HADAPTION5X2HNO)
153   FORMAT(21X8HADAPTION5X3HYES)
154   FORMAT(8X21HHALF-STEP INTEGRATION5X2HNO)
155   FORMAT(8X21HHALF-STEP INTEGRATION5X3HYES)
156   FORMAT(10X19HPUNCHED CARD OUTPUT5X2HNO)
157   FORMAT(10X19HPUNCHED CARD OUTPUT5X3HYES)
158   FORMAT(14X15HNUMBER OF PLOTS5XI3)
162   FORMAT(11X18HLOW LEVEL CONTOURS5X2HNO)
163   FORMAT(11X18HLOW LEVEL CONTOURS5X3HYES)
164   FORMAT(3X26HRESCALE FINAL CONTOUR PLOT5X2HNO)
165   FORMAT(3X26HRESCALE FINAL CONTOUR PLOT5X3HYES)
168   FORMAT(20X9HBEAMSHAPE5X*UNIFORM CIRCLE - OCCULTED RADIUS =*F5.2)
C
C*****************************************************************************
C
C    DEFINE AND STORE COMPARATIVE PHYSICAL DATA
C
       C13=1.0/3.0
       W2=WIDTH*WIDTH
       WL=(2.0*PI/WN)*1.0E4
       BDI=BDI*(10.6/WL)**2
       RHT=1.0/HT
       POUT=ENERGY*RHT*EJTKJ
       DAREA=HX*HY
       CAS=SQRT(ALPHA*F*F*ENERGY**3/(9.0E-6*(PI*WIDTH*BDI)**2))
       CTP=3.0E-3*SQRT(W2/(ALPHA*F*F*ENERGY))
       CPCR=(3.0E-8*BDI*BDI/((GAMMA-1.0)*ALPHA*F*F))**C13*RHT*PI*W2
       ZFINAL=ZF*CTK*0.999999
       ZZF=PNZ/PNK
       TPULSE=HT
       NP=10
C
      DO 54 I=1,NP
       ZNM=FLOAT(I-1)/FLOAT(NP-1)+1.0E-3
       ZETA=ZNM/PNK
       Z=ZNM*F*CTK
       D=ZETA*ZETA+(1.0-ZNM)**2
       AV=W2*D
       EX=EXP(-PNALF*ZNM)
       SREX=SQRT(EX)
       AS4=CAS*ZNM*SREX**3
       AS2=SQRT(AS4)
       AR=AS2/AV
       PCR=CPCR*AS4**C13/EX
       TP=CTP*AS2/(SREX*ZNM+1.0E-60)
       TPULSE=AMIN1(TPULSE, TP)
       TS=PCR*TP/POUT
       SI=0.75*PCR*EX/(PI*AV)
       IF (Z .NE. 0.0) GO TO 10
       TCOR1=1.0
       TCOR2=1.0
```

```
      GO TO 12
10    DTURB1=DTURBF(1.0E-15)
      TCOR1=D/DTURB1
      DTURB2=DTURBF(1.0E-14)
      TCOR2=D/DTURB2
12    CONTINUE
      BOUT(1,I)=Z
      BOUT(2,I)=AS2
      BOUT(3,I)=AR
      BOUT(4,I)=TP
      BOUT(5,I)=PCR
      BOUT(6,I)=TS
      BOUT(7,I)=SI
      BOUT(8,I)=TCOR1
      BOUT(9,I)=TCOR2
 54   CONTINUE
C
C****************************************************************************
C
C     DEFINE INITIAL AMPLITUDES AT APERTURE
C
C     TRUNCATED OR INFINITE GAUSSIAN
C     UNIFORM CIRCLE OR SQUARE
C
C     TRUNCATED GAUSSIAN IS TRUNCATED AT 1/E INTENSITY RADIUS
C     OR R(TRUN.)=1.414*A
C
      XZERO=-(NX-1)*HX/2.
      YZERO=-(NY-1)*HY/2.
      NXM=NX-1
      NYM=NY-1
      DKAREA=NX*NY*DAREA
      DO 64 J=1,NY
      Y=(J-1)*HY +YZERO
      G2(J)=1.0-Y*Y
      DO 64 I=1,NX
      X=(I-1)*HX+XZERO
      IF (J .EQ. 1) G1(I)=1.0-X*X
      SSQ=X*X+Y*Y
C
C     DEFINE GAUSSIAN AMPLITUDE
C
      IF (NBM.GE.2) GO TO 300
      REAL=EXP(-0.5*SSQ)
      IF (NBM.EQ.1.AND.SSQ.GT.2.0) REAL=0.0
      GO TO 350
C
C     DEFINE UNIFORM CIRCLE AMPLITUDE
C
300   IF (NBM.GT.2) GO TO 310
      REAL=1.0
      IF (SSQ.GT.1.0) REAL=0.0
      GO TO 350
C
C     DEFINE SQUARE APERTURE
C
310   IF (NBM.GT.3) GO TO 320
      IF (ABS(X).LE.1.0.AND.ABS(Y).LE.1.0.AND.NBM.EQ.3) REAL=1.0
      IF (ABS(X).GT.1.0.OR.ABS(Y).GT.1.0.AND.NBM.EQ.3) REAL=0.0
      GO TO 350
C
C     DEFINE OCCULTED UNIFORM CIRCLE
C
320   REAL=1.0
      IF (SSQ.GT.1.0) REAL=0.0
      IF (SQRT(SSQ).LE.ROCULT) REAL=0.0
```

```
C
C    LOAD INITIAL ARRAY
C
350    CONTINUE
        A1(I,J)=CMPLX(REAL,0.0)
 64    CONTINUE
C
C    FIND NORMALIZATION FACTOR
C
        CALL INTENS(A1, .FALSE.)
        RNORM=0.0
       DO 66 J=1,NY
       DO 66 I=1,NX
        RNORM=RNORM+TENS(I,J)
66     CONTINUE
        RRNORM=1.0/SQRT(RNORM*DAREA)
C
C*******************************************************************************
C
C    INITIALIZE CONSTANTS AND PARAMETERS USED LATER IN PROGRAM
C
       NBUF=2*NXDIM*NY
       NXY=2*NXDIM*NYDIM

       NY2=NY/2

        PV(I)=2.0*PI*(I-1)/(NX*HX)

       ILO=NX2+1
       DO 70 I=ILO,NX
        PV(I)=2.0*PI*(I-1-NX)/(NX*HX)
70      PV(I)=PV(I)*PV(I)
       DO 72 J=1,NY
        PHASE2(J)=0.5*PV(J)
       DO 72 I=1,NX
C
C    NORMALIZE AMPLITUDES IN THIS LOOP
C
        A1(I,J)=RRNORM*A1(I,J)
        A2(I,J)=A1(I,J)
        EPS(I,J)=0.0
        IF (J .EQ. 1) PHASE1(I)=0.5*PV(I)
72     CONTINUE
       DO 74 J=1,NY
74      EPSO(1,J)=0.0
C
C    STORE X AND Y SLICES AT APERTURE
C
       IPX=NX/2
       IPY=NY/2
       DO 420 I=1,64
       AIN(1,I)=A1(IPX,I)
420    AIN(2,I)=A1(I,IPY)
C
        RI63MX=0.0
        RIMXMX=0.0
        APMN=1.0E10
        HZMN=1.0F10
        HZ=0.0
        CON1=0.5*(1.0-1.0/SQRT(3.0))
        CON2=0.5*(1.0+1.0/SQRT(3.0))
       M2(1)=LOGF(1.*NX)/LOGF(2.)+0.5
       M2(2)=LOGF(1.*NY)/LOGF(2.)+0.5
       M2(3)=0
       CALL SETUP (0,M2,SV1,SV2,0,IFERR)
C
```

```
C    DEFINE CONTOUR LEVELS
C
       IF (NCT.NE.0) GO TO 210
       CTLVL=.9
       DO 200 I=1,9
       CONMIN(I)=CTLVL
200    CTLVL=CTLVL-.1
       CONMIN(10)=0.05
       GO TO 250
210    CONMIN(1)=0.50
       DO 220 I=2,10
220    CONMIN(I)=CONMIN(I-1)*0.5
250    CONTINUE
       TLAST=TIMELEFT(0)
C
C    INITIALIZE  PROGRAM PARAMETERS
C
        Z=0.0
        ZZ=0.0
        HCZ1N=0.0
        HCZ2N=0.0
       ZNM=0.
       D=1.
        D2=1.0
        SQRT8=SQRT(8.0)
       DO 80 I=1,2
        D10(I)=1.0
        D20(I)=1.0
        RD10(I)=1.0
        RD20(I)=1.0
        ALPH10(I)=-0.5*PNK
        ALPH20(I)=-0.5*PNK
        BET10(I)=0.0
        SRTD10(I)=1.0
        XCEN0(I)=0.0
        DALPH1(I)=0.0
        DALPH2(I)=0.0
        DBET1(I)=0.0
80     CONTINUE
        SRTD1=1.0
        SRTD2=1.0
        RSRD12=1.0
        NFLAG=0
C
C
C    PLOT INITIAL INTENSITY DISTRIBUTION AT APERTURE
C
       IF (NPLOT.LT.4) GO TO 260
       CALL SYMBOL(0.0,8.0,0.14,3HZ =,0.0,3)
       CALL NUMBER(0.36,8.0,0.14,Z,0.0,4HF8.5)
       CALL SYMBOL(1.32,8.0,0.14,4H KM ,0.0,4)
       CALL LABEL(0.0,4.0)
       CALL PLOT(2.5,0.0,-3)
       XCENTER=5.0
       CALL SYMBOL(XCENTER,5.0,0.14,3,0.0,-1)
       XMIN=XZERO*WIDTH
       YMIN=YZERO*WIDTH
       CALL TOPOGRAF(TENS,NXDIM,NYDIM,NX,NY,0.0,0.0,11,10.0,10.0,ITENS,
      1 XMIN,HX,4HF6.1,4HX CM,+4,YMIN,HY,4HF6.1,4HY CM,4)
260    RETURN
       END
```

```
      SUBROUTINE ADVNCE(A,NS)
C
C    THIS SUBROUTINE FOURIER TRANSFORMS PHI(X,Y,Z) TO PHI(K1,K2,Z),
C    AND ADVANCES THE SOLUTION BY APPLYING THE PHASE CHANGE
C
C    PHI(K1,K2,Z+HZ)=PHI(K1,K2,Z)*EXP(0.5*I*(K1**2*HZ/D1+K2**2*HZ/D2))
C
C    AND THEN TRANSFORMS BACK TO REAL SPACE
C
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
     COMMON /AAA/ EPS(64,64),EPSO(64,64),AOUT(11,99),BOUT(9,10),
    . AIN(2,64),DALPH1(2),DALPH2(2),DBET1(2),ALPH10(2),ALPH20(2),
    .   BET10(2), D10(2), D20(2), RD10(2), RD20(2), SRTD10(2), XCEN0(2)
      COMMON /BBB/ TENS(64,64), G1(64), G2(64), PHASE1(64), PHASE2(64),
    . CONMIN(10),M2(3),SV1(64),SV2(64),PARM(80)
      COMMON /SINGLS/ F, PNA, PNALF, PNK, PNO, PNS, PND, PNZ, HX, HY,
    .   HZ, Z, ZZ,ZZF, ZNM, ZFINAL, XZERO, YZERO, WIDTH, ALPHA, WN,
    .   VODT, OMDT, HT, ENERGY, ALPHAC, CS, REFRAC, GAMMA, ETJ, CTK,
    .  EJTKJ, RHT, POUT, DAREA, W2, TS, TPULSE, AS2,PCR,SI,TCOR1,TCOR2,
    .   Z1, RI63MX, Z2, RIMXMX, Z3, APMN, Z4, HZMN, DKAREA, TENSMX,
    .   EX, PHIMX, EPSMX, ERRMX, DGMX, R1, BDIMAX, VTERM, PHIMXX,HZNMS,        14
    .   PI, IMAX, JMAX, NX, NY, NAD, NX2, NY2, NXY, NXDIM, NYDIM, NPT,
    .   IPLOT,NITER,NBUF,NXM,NYM,NMS,NFLAG,D,D1,D2,P1,P2,SRTD1,SRTD2,
    .   RSRD12,XCEN,TLAST,SQRT8,PNDO,GCONO,GCON,BDI,HCZ1O,HCZ2O,HCZ1N,
    .   HCZ2N,HCZ12,ALPH1,ALPH2,BET1,CON1,CON2,HZO,HZN,EXO,EXN,WT1,WT2
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      COMPLEX A(64,64)
C
C    DEFINE PARAMETERS FOR PHASE TRANSFORMATION
C
      HZ1=CON1*HZN
      HZ2=CON2*HZN
      ZD11=(HZ1+HZO)*RD10(NS)
      ZD12=(HZ2+HZO)*RD10(NS)
      ZD21=(HZ1+HZO)*RD20(NS)
      ZD22=(HZ2+HZO)*RD20(NS)
      D11=D10(NS)*  ((1.0+2.0*ALPH10(NS)*  ZD11)**2+ZD11*ZD11)
      D12=D10(NS)*  ((1.0+2.0*ALPH10(NS)*  ZD12)**2+ZD12*ZD12)
      D21=D20(NS)*  ((1.0+2.0*ALPH20(NS)*  ZD21)**2+ZD21*ZD21)
      D22=D20(NS)*  ((1.0+2.0*ALPH20(NS)*  ZD22)**2+ZD22*ZD22)
      RD11=1.0/D11
      RD12=1.0/D12
      RD21=1.0/D21
      RD22=1.0/D22
      HCZ1N=0.5*HZN*(RD11+RD12)
      HCZ2N=0.5*HZN*(RD21+RD22)
      RSRDS1=SQRT(RD11*RD21)
      RSRDS2=SQRT(RD12*RD22)
      HCZ12=0.5*HZN*(RSRDS1+RSRDS2)
C
C    WHEN NMS=0, P2=0
C
      WT1=P2*(HCZ12-HZ1*RSRDS1-HZ2*RSRDS2)
      WT2=HCZ12-WT1
C
C    RESORT ARRAY IF NX LT 64
C
      IF (NX .EQ. NXDIM) GO TO 1155
     DO 115 J=1,NY
     DO 115 I=1,NX
115   A(I+(J-1)*NX)=A(I,J)
1155 CONTINUE
```

```
C
C     PERFORM FOURIER TRANSFORM - TO K-SPACE
C
         CALL FASTFOUR(A(1,1), M2, SV1, SV2, -1, IFERR)
         IF (NX .LT. NXDIM) GO TO 10
C
C     PLOT FOURIER TRANSFORM OF INTENSITY DISTRIBUTIONS AT
C     APERTURE AND Z-FINAL
C
         IF (NITER .EQ. 1 .AND. NS .EQ. 1) CALL INTENS(A, .TRUE.)
        IF (ZZ+HZN+HZNMS.EQ.ZZF) CALL INTENS(A,.TRUE.)
10      CONTINUE
C
C     APPLY PHASE CHANGE TO ADVANCE THE CALCULATIONS
C
        DO 12 J=1,NY
        DO 12 I=1,NX
         JT=(J-1)*NX+I
         PHI=P1*((HCZ10+HCZ1N)*PHASE1(I)+(HCZ20+HCZ2N)*PHASE2(J))
12       A(JT)=A(JT)*CMPLX(COS(PHI),-SIN(PHI))
C
C     PERFORM FOURIER TRANSFORM - TO REAL-SPACE
C
         CALL FASTFOUR(A(1,1), M2, SV1, SV2,  1, IFERR)
C
C     RESORT ARRAY IF NX LT 64
C
         IF (NX .EQ. NXDIM) GO TO 1165
        DO 116 J=1,NY
        DO 116 I=1,NX
116      A(I,J)=A(I+(J-1)*NX)
 1165   RETURN
        END
```

```
      SUBROUTINE DENS(A, B, ZCOORD, ND1, ND2, LD)
C
C     THIS SUBROUTINE APPLYS THE PHASE CHANGE
C
C     (1-X**2)/D1+(1-Y**2)/D2-3N(GAMMA-1)*K**2*E/CS**2/SQRT(D1*D2)*
C     SUM(PHI(X-XP,Y,Z))**2
C
C     THIS CONVERTS PSI (OR A) TO PHI
C
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      COMMON /AAA/ EPS(64,64),EPSO(64,64),AOUT(11,99),BOUT(9,10),
     . AIN(2,64),DALPH1(2),DALPH2(2),DBET1(2),ALPH10(2),ALPH20(2),
     .   BET10(2), D10(2), D20(2), RD10(2), RD20(2), SRTD10(2), XCENO(2)
      COMMON /BBB/ TENS(64,64), G1(64), G2(64), PHASE1(64), PHASE2(64),
     . CONMIN(10),M2(3),SV1(64),SV2(64),PARM(80)
      COMMON /SINGLS/ F, PNA, PNALF, PNK, PNO, PNS, PND, PNZ, HX, HY,
     .   HZ, Z, ZZ,ZZF, ZNM, ZFINAL, XZERO, YZERO, WIDTH, ALPHA, WN,
     .   VODT, OMDT, HT, ENERGY, ALPHAC, CS, REFRAC, GAMMA, ETJ, CTK,
     .  EJTKJ, RHT, POUT, DAREA, W2, TS, TPULSE, AS2,PCR,SI,TCOR1,TCOR2,
     .   Z1, RI63MX, Z2, RIMXMX, Z3, APMN, Z4, HZMN, DKAREA, TENSMX,
     .   EX, PHIMX, EPSMX, ERRMX, DGMX, R1, BDIMAX, VTERM, PHIMXX,HZNMS,     14
     .   PI, IMAX, JMAX, NX, NY, NAD, NX2, NY2, NXY, NXDIM, NYDIM, NPT,
     .   IPLOT,NITER,NBUF,NXM,NYM,NMS,NFLAG,D,D1,D2,P1,P2,SRTD1,SRTD2,
     .   RSRD12,XCEN,TLAST,SORT8,PNDO,GCONO,GCON,BDI,HCZ1O,HCZ2O,HCZ1N,
     .   HCZ2N,HCZ12,ALPH1,ALPH2,BET1,CON1,CON2,HZO,HZN,EXO,EXN,WT1,WT2
      COMMON /OUTS/ NBM,SCLFAC,NRS,NPM,NCW,NEXIT,NPLOT,NPUNCH
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      COMPLEX A(64,64), B(64,64)
      LOGICAL LD
      DIMENSION TENS(10), EPSO(10), HZSAVE(2)
C
C     INITIALIZE Z-STEP ON FIRST CALL
C
      IF (LD) GO TO 41
      HZN=0.0
      HZNMS=0.0
      HZSAVE(1)=0.0
      HZSAVE(2)=0.0
      P1=0.5
      P2=0.0
     P3=1.0
      EXN=1.0
C
C     ZZF = ZFINAL / K * WIDTH**2
C
     HZMX=0.10*ZZF
     IF (NMS.EQ.0) HZMX=0.5*HZMX
      HZMINI=1.0E-4*ZZF
     IF (NMS.EQ.0) HZMINI=0.5*HZMINI
41    ZCOORD=ZCOORD+HZSAVE(ND1)
C
C     ZZ = Z(CM) / K * WIDTH**2
C
42    ZZ=ZCOORD
C
C     ZNM = Z / F
C
      ZNM=ZZ*PNK
C
C     Z = Z(KM)
C
      Z=ZNM*F*CTK+1.0E-6
C
      D=ZZ*ZZ+(1.0-ZNM)**2
      ZDD1=HZ*RD10(ND1)
```

UNCLASSIFIED

```
      D1=D10(ND1)*((1.0+2.0*ALPH10(ND1)*ZDD1)**2+ZDD1*ZDD1)
      SRTD1=SQRT(D1)
C
C    VTERM = DISTANCE BETWEEN PULSES / DISTANCE BETWEEN GRID POINTS
C
      VTERM=2.0*(1.0+PNS*ZNM)/(HX*SRTD1*PNO)
      IF(VTERM.LT.0) GO TO 45
      EPSMX=0.0
      JC=IFIX(1.0/VTERM)+1
      DC=FLOAT(JC-2)
      IF (NCW.EQ.1) GO TO 200
      IF (DC) 48,49,50
C
C    EXIT OPTIONS
C
45    NPUNCH=0
      NEXIT=1
      PRINT 100, Z
100   FORMAT(//2X, *AT Z=*, F6.4, * KM, A DEAD ZONE IS PRESENT IN THE C
     .ALCULATION*)
      CALL OUTPUT(.TRUE.)
      STOP
C
46    PRINT 101, Z
101   FORMAT ( // 2X *AT Z= * F6.4 * KM, THERE ARE MORE THAN 10  PULSES
     . PER CELL PRESENT IN THE CALCULATION*)
      STOP
C
47    NPUNCH=0
      NEXIT=1
      PRINT 103, Z
103   FORMAT( // 2X *AT Z= * F7.4 * KM, THE CALCULATED HZ IS SMALLER TH
     .AN THE MINIMUM ALLOWED VALUE* )
      CALL OUTPUT(.TRUE.)
      STOP
C
C*****************************************************************************
C
C    SUM THE INTENSITY ACROSS THE GRID FOR MULTI-PULSE
C    INTEGRATE THE INTENSITY ACROSS THE GRID FOR CW
C
C
C    LESS THAN ONE PULSE PER CELL
C
48    I1=VTERM
      I11=I1+1
      F1=I11-VTERM
      F2=1.0-F1
      DO 4 J=1,NY
      DO 4 I=2,NX
      IF(I-I11.GE.1) GO TO 44
      EPSO(I,J)=EPS(I,J)
      EPS(I,J)=0.0
      GO TO 4
44    EPSO(I,J)=EPS(I,J)
      EPS(I,J)=F1*(TENS(I-I1,J)+EPS(I-I1,J))+
     .   F2*(TENS(I-I11,J)+EPS(I-I11,J))
      IF (TENS(I,J) .GT. 0.05*TENSMX) EPSMX=AMAX1(EPSMX,EPS(I,J))
4     CONTINUE
      GO TO 51
C
C    ONE TO TWO PULSES PER CELL
C
49    UTERM=2.0*VTERM
      F1=2.0-UTERM
      F2=1.0-F1
```

```
      DO 5 J=1,NY
       EPS1=0.0
       TEN1=0.5*(TENS(1,J)+TENS(2,J))
       EPSO(2,J)=EPS(2,J)
       EPS(2,J)=F1*TEN1+F2*TENS(1,J)
       EPSMX=AMAX1(EPS(2,J), EPSMX)
      DO 5 I=3,NX
       EPS1=F1*(TENS(I-1,J)+EPS(I-1,J))+F2*(TEN1+EPS1)
       TEN1=0.5*(TENS(I-1,J)+TENS(I,J))
       EPSO(I,J)=EPS(I,J)
       EPS(I,J)=F1*(TEN1+EPS1)+F2*(TENS(I-1,J)+EPS(I-1,J))
       IF (TENS(I,J) .GT. 0.05*TENSMX) EPSMX=AMAX1(EPSMX,EPS(I,J))
5     CONTINUE
       GO TO 51
C
C    MORE THAN TWO PULSES PER CELL
C
50     IF (JC .GT. 10) GO TO 46
       UTERM=FLOAT(JC)*VTERM
       F1=2.0-UTERM
       F2=1.0-F1
      DO 6 J=1,NY
```

UNCLASSIFIED

```
      I=2
      EPSO(1)=0.0
      EPSO(2)=0.0
      TENO(1)=TENS(I-1,J)
      TENO(2)=(TENS(I,J)+(FLOAT(JC)-1.0)*TENS(I-1,J))/FLOAT(JC)
      DO 70 JJ=3,JC
      FJ=FLOAT(JJ-1)/FLOAT(JC)
      TENO(JJ)=FJ*TENS(I,J)+(1.0-FJ)*TENS(I-1,J)
      EPSO(JJ)=F1*(TENO(JJ-1)+EPSO(JJ-1))+F2*(TENO(JJ-2)+EPSO(JJ-2))
70    CONTINUE
      EPSO(I,J)=EPS(I,J)
      EPS(I,J)=F1*(TENO(JC)+EPSO(JC))+F2*(TENO(JC-1)+EPSO(JC-1))
      EPSMX=AMAX1(EPS(I,J), EPSMX)
      DO 6 I=3,NX
      EPSO(1)=EPS(I-1,J)
      TENO(1)=TENS(I-1,J)
      EPSO(2)=F1*(TENO(1)+EPSO(1))+F2*(TENO(JC)+EPSO(JC))
      TENO(2)=(TENS(I,J)+(FLOAT(JC)-1.0)*TENS(I-1,J))/FLOAT(JC)
      DO 71 JJ=3,JC
      FJ=FLOAT(JJ-1)/FLOAT(JC)
      TENO(JJ)=FJ*TENS(I,J)+(1.0-FJ)*TENS(I-1,J)
      EPSO(JJ)=F1*(TENO(JJ-1)+EPSO(JJ-1))+F2*(TENO(JJ-2)+EPSO(JJ-2))
71    CONTINUE
      EPSO(I,J)=EPS(I,J)
      EPS(I,J)=F1*(TENO(JC)+EPSO(JC))+F2*(TENO(JC-1)+EPSO(JC-1))
      IF (TENS(I,J) .GT. 0.05*TENSMX) EPSMX=AMAX1(EPSMX,EPS(I,J))
6     CONTINUE
      GO TO 51
C
C    COMPUTE CW INTEGRAL
C
200   DO 110 J=1,NY
      EPSO(1,J)=EPS(1,J)
      EPS(1,J)=0.5*HX*TENS(1,J)*WIDTH/(VODT*(1.+PNS*ZNM))
     1      *SQRT(D1)
      DO 110 I=2,NX
      EPSO(I,J)=EPS(I,J)
      EPS(I,J)=EPS(I-1,J)+0.5*HX*(TENS(I,J)+TENS(I-1,J))*WIDTH/
     1 (VODT*(1.+PNS*ZNM))
     1      *SQRT(D1)
      IF (TENS(I,J).GT.0.05*TENSMX) EPSMX=AMAX1(EPSMX,EPS(I,J))
110   CONTINUE
C
C*************************************************************************
```

```
C
51      EXO=EXN
        EX=EXP(-PNALF*ZNM)
        EXN=EX
        GCON=GCONO*EX
        IF (LD) CALL VTRANS(A,ND1)
        HZO=HZN
        HCZ1O=HCZ1N
        HCZ2O=HCZ2N
C
C     CALCULATE THE Z-INTEGRALS WHEN NMS = 0
C
        IF (NMS.NE.0) GO TO 52
        HZNMS=HZO
        HZ1=CON1*HZO
        HZ2=CON2*HZO
        ZD11=HZ1*RD10(ND1)
        ZD12=HZ2*RD10(ND1)
        ZD21=HZ1*RD20(ND1)
        ZD22=HZ2*RD20(ND1)
        D11=D10(ND1)*((1.0+2.0*ALPH10(ND1)*ZD11)**2+ZD11*ZD11)
        D12=D10(ND1)*((1.0+2.0*ALPH10(ND1)*ZD12)**2+ZD12*ZD12)
        D21=D20(ND1)*((1.0+2.0*ALPH20(ND1)*ZD21)**2+ZD21*ZD21)
        D22=D20(ND1)*((1.0+2.0*ALPH20(ND1)*ZD22)**2+ZD22*ZD22)
        RD11=1.0/D11
        RD12=1.0/D12
        RD21=1.0/D21
        RD22=1.0/D22
        RSRDS1=SQRT(RD11*RD21)
        RSRDS2=SQRT(RD12*RD22)
C
C     THE 3 Z-INTEGRALS
C
        HCZ1O=0.5*HZO*(RD11+RD12)
        HCZ2O=0.5*HZO*(RD21+RD22)
        HCZ12=0.5*HZO*(RSRDS1+RSRDS2)
        WT2=HCZ12
C
C     COMPUTE NEW Z INCREMENT IN Z/KA**2 UNITS
C
52      HZN=P3*AMIN1(0.04*D1, 0.04*D2, PHIMXX/
     .     (RSRD12*GCON*(EPSMX+1.0E-50)))
        IF (HZN.GT.HZMX) HZN=HZMX
        IF (HZN .LT. HZMINI) GO TO 47
        IF (HZN .GT. ZZF-ZZ-HZNMS) HZN=ZZF-ZZ-HZNMS
        HZ=HZO+HZN
        HZSAVE(ND2)=HZ
C
C     COMPUTE THE THREE Z-INTEGRALS
C
C     DO ONLY ON FIRST CALL
C
        IF (LD) GO TO 54
        IF (NMS.EQ.0) P3=0.5
        HZ1=CON1*HZ
        HZ2=CON2*HZ
        ZD11=HZ1*RD10(ND2)
        ZD12=HZ2*RD10(ND2)
        ZD21=HZ1*RD20(ND2)
        ZD22=HZ2*RD20(ND2)
        D11=D10(ND2)*((1.0+2.0*ALPH10(ND2)*ZD11)**2+ZD11*ZD11)
        D12=D10(ND2)*((1.0+2.0*ALPH10(ND2)*ZD12)**2+ZD12*ZD12)
        D21=D20(ND2)*((1.0+2.0*ALPH20(ND2)*ZD21)**2+ZD21*ZD21)
        D22=D20(ND2)*((1.0+2.0*ALPH20(ND2)*ZD22)**2+ZD22*ZD22)
        RD11=1.0/D11
        RD12=1.0/D12
```

```
      RD21=1.0/D21
      RD22=1.0/D22
      RSRDS1=SQRT(RD11*RD21)
      RSRDS2=SQRT(RD12*RD22)
      HCZ10=0.5*HZ*(RD11+RD12)
      HCZ20=0.5*HZ*(RD21+RD22)
      HCZ12=0.5*HZ*(RSRDS1+RSRDS2)
      WT1=0.0
      WT2=HCZ12
C
C     COMPUTE THE PHASE CHANGE IN THIS LOOP WHEN NMS NE 0
C
54    CONTINUE
```

```
      PHIMX=0.0
      IF (NMS .EQ. 0) GO TO 80
      DO 55 J=1,NY
      Y=YZERO+FLOAT(J-1)*HY
      DO 55 I=1,NX
      X=XZERO+FLOAT(I-1)*HX
      GNEW=GCONO*(WT1*EXO*EPSO(I,J)+WT2*EXN*EPS(I,J))
      PHI=0.5*P1*(HCZ10*G1(I)+HCZ20*G2(J)-GNEW)
      PHI=PHI-X*X*DALPH1(ND2)-Y*Y*DALPH2(ND2)-X*DBE11(ND2)
      B(I,J)=B(I,J)*CMPLX(COS(PHI), SIN(PHI))
      IF (TENS(I,J) .LT. 0.05*TENSMX) GO TO 55
      PHIMX=AMAX1(PHIMX, 0.5*GNEW)
 55   CONTINUE
      GO TO 60
C
C   COMPUTE THE PHASE CHANGE IN THIS LOOP WHEN NMS = 0
C
80    DO 85 J=1,NY
      Y=YZERO+FLOAT(J-1)*HY
      DO 85 I=1,NX
      X=XZERO+FLOAT(I-1)*HX
      GNEW=GCONO*(WT1*EXO*EPSO(I,J)+WT2*EXN*EPS(I,J))
      PHI=0.5*P1*(HCZ10*G1(I)+HCZ20*G2(J)-GNEW)
      PHI=PHI-X*X*DALPH1(ND2)-Y*Y*DALPH2(ND2)-X*DBE11(ND2)
      A(I,J)=A(I,J)*CMPLX(COS(PHI), SIN(PHI))
      IF (TENS(I,J) .LT. 0.05*TENSMX) GO TO 85
      PHIMX=AMAX1(PHIMX, 0.5*GNEW)
85    CONTINUE
 60   RETURN
      END
```

```
      SUBROUTINE INTENS(A, LI)
C
C   THIS SUBROUTINE TAPERS THE BOUNDARIES OF THE COMPUTATIONAL
C   GRID TO ZERO, FINDS THE INTENSITY AT EACH GRID POINT
C   AND PLOTS THE CONTOURS OF THE FOURIER TRANSFORMED MATRIX.
C
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
     COMMON /AAA/ EPS(64,64),EPSO(64,64),AOUT(11,99),BOUT(9,10),
    . AIN(2,64),DALPH1(2),DALPH2(2),DBET1(2),ALPH10(2),ALPH20(2),
    .   BET10(2), D10(2), D20(2), RD10(2), RD20(2), SRD10(2), XCEN0(2)
      COMMON /BBB/ TENS(64,64), G1(64), G2(64), PHASE1(64), PHASE2(64),
    . CONMIN(10),M2(3),SV1(64),SV2(64),PARM(80)
      COMMON /SINGLS/ F, PNA, PNALF, PNK, PNO, PNS, PND, PNZ, HX, HY,
    .   HZ, Z, ZZ,ZZF, ZNM, ZFINAL, XZERO, YZERO, WIDTH, ALPHA, WN,
    .   VODT, OMDT, HT, ENERGY, ALPHAC, CS, REFRAC, GAMMA, ETO, CTK,
    .  EJTKJ, RHT, POUT, DAREA, W2, TS, TPULSE, AS2,PCR,SI,TCOR1,TCOR2,
    .   Z1, RI63MX, Z2, RIMXMX, Z3, APMN, Z4, HZMN, DKAREA, TENSMX,
    . . EX, PHIMX, EPSMX, ERRMX, DGMX, R1, BDIMAX, VTERM, PHIMXX,HZNMS,    14
    .   PI, IMAX, JMAX, NX, NY, NAD, NX2, NY2, NXY, NXDIM, NYDIM, NPI,
    .   IPLOT,NITER,NBUF,NXM,NYM,NMS,NFLAG,D,D1,D2,P1,P2,SRID1,SRID2,
    .   RSRD12,XCEN,TLAST,SQRT8,PND0,GCON0,GCON,BDI,HCZ10,HCZ20,HCZ1N,
    .   HCZ2N,HCZ12,ALPH1,ALPH2,BET1,CON1,CON2,HZ0,HZN,EX0,EXN,WT1,WT2
     COMMON /OUTS/ NBM,SCLFAC,NRS,NPM,NCW,NEXIT,NPLOT,NPUNCH
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      COMPLEX A(64,64)
      LOGICAL LI
C
C   TAPER BOUNDARY VALUES TO ZERO
C
     IF (LI) GO TO 120
     DO 100 I=1,NX
     A(I,1)=0.0
     A(I,NY)=0.0
     A(I,2)=A(I,2)*0.5
100  A(I,(NY-1))=A(I,(NY-1))*0.5
     DO 110 J=1,NY
     A(1,J)=0.0
     A(NX,J)=0.0
     A(2,J)=A(2,J)*0.5
110  A((NX-1),J)=A((NX-1),J)*0.5
120  CONTINUE
C
C   COMPUTE THE INTENSITY AT EACH GRID POINT AND LOCATE THE MAXIMUM
C
      TENSMX=0.0
     DO 9 J=1,NY
     DO 9 I=1,NX
      TENS(I,J)=A(I,J)*CONJG(A(I,J))
      IF (TENS(I,J) .LE. TENSMX) GO TO 9
      IMAX=I
      JMAX=J
      TENSMX=TENS(I,J)
 9   CONTINUE
C
C   RETURN IF NOT PLOTTING FOURIER TRANSFORMS
C
      IF (.NOT. LI) GO TO 15
C
C   RESORT ARRAY WHEN PLOTTING FOURIER TRANSFORMS
C
     DO 16 J=1,NY2
     DO 18 I=1,NX2
      HOLD=TENS(I,J)
      TENS(I,J)=TENS(I+NX2,J+NY2)
      TENS(I+NX2,J+NY2)=HOLD
18   CONTINUE
```

NRL Report 8074

UNCLASSIFIED

SSPARAMA: A Nonlinear, Wave Optics Multipulse (and CW) Steady-State Propagation Code with Adaptive Coordinates

K. G. WHITNEY AND G. L. MADER

Science Applications Inc.
Arlington, Virginia

and

P. B. ULRICH

Optical Radiation Branch
Optical Sciences Division

February 10, 1977

NAVAL RESEARCH LABORATORY
Washington, D.C.

SECURITY CLASSIFICATION OF THIS PAGE (*When Data Entered*)

UNCLASSIFIED

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: NRL Report 8074	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (*and Subtitle*): SSPARAMA: A NONLINEAR, WAVE OPTICS MULTIPULSE (AND CW) STEADY-STATE PROPAGATION CODE WITH ADAPTIVE COORDINATES	5. TYPE OF REPORT & PERIOD COVERED: Final report on one phase of a continuing NRL problem
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(*s*): K. G. Whitney, G. L. Mader, and P. B. Ulrich	8. CONTRACT OR GRANT NUMBER(*s*)
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Naval Research Laboratory, Washington, D.C. 20375	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: NRL Problem R05-31, Project SAA-76
11. CONTROLLING OFFICE NAME AND ADDRESS: Naval Sea Systems Command (PMS-405), Washington, D.C. 20362	12. REPORT DATE: February 10, 1977
	13. NUMBER OF PAGES: 80
14. MONITORING AGENCY NAME & ADDRESS(*if different from Controlling Office*)	15. SECURITY CLASS. (*of this report*): Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (*of this Report*)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (*of the abstract entered in Block 20, if different from Report*)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (*Continue on reverse side if necessary and identify by block number*)

Laser propagation
Nonlinear optics
Atmospheric optics
Computer simulation

20. ABSTRACT (*Continue on reverse side if necessary and identify by block number*)

This report describes the numerical procedures that are used to calculate the steady-state atmospheric propagation and thermal blooming of either a train of high-energy laser pulses or a CW beam. The calculation is performed through the aid of a sequence of coordinate and variable transformations that enable one to adapt the scale and location of the transverse coordinate system to the amount of diffraction or thermal blooming taking place during beam propagation. A description is also given of the input parameters and diagnostics provided in the calculation along with a listing of the computer program.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (*When Data Entered*)

CONTENTS

SSPARAMA: A NONLINEAR, WAVE OPTICS MULTIPULSE (AND CW) STEADY-STATE PROPAGATION CODE WITH ADAPTIVE COORDINATES

INTRODUCTION

Several methods of propagating CW high-energy laser beams through the atmosphere have been reported previously [1,2]. This report will describe a method for propagating multiply pulsed laser beams in a nonlinear atmosphere by adapting the coordinate system to the amount of thermal blooming. This technique increases the accuracy of thermal-blooming calculations and extends the capability of the code in the case of extreme beam distortion.

The computer code SSPARAMA calculates the steady-state intensity pattern of a train of high-energy laser pulses propagating through the atmosphere in the presence of thermal blooming. Steady state is achieved when enough equally spaced, equal-energy pulses have been propagated for transients in air heating to have died out. In the steady state a single pulse will propagate in an atmosphere that has been heated by many preceding pulses which have the same energy distribution as the pulse one is calculating. The pulse widths are assumed to be short compared to the sound transit time across the face of the beam, so that self-blooming will not take place. Blooming occurs only as a result of air heating by preceding pulses. However, to avoid problems of plasma formation, the pulse width must be sufficiently long that the critical intensity for air breakdown is not exceeded. Finally, as the pulse is propagated from one coordinate plane to another, coordinate transformations are performed to insure that the transverse scale lengths are adapted to the amount of thermal blooming induced on the pulse train by the negative lensing influence of the heated atmosphere.

Another requirement for steady-state propagation is that a cooling mechanism exist for removing heated air from the path of the beam. In SSPARAMA, cooling is provided either by a wind moving perpendicular to the propagation direction or by beam sluing about an axis in the aperture plane perpendicular to both the wind and the propagation directions. The steady-state density changes $\Delta\rho$ introduced in the path of a given pulse by energy absorption from all preceding pulses can then be expressed as [3]

$$\Delta\rho = -\frac{\gamma - 1}{c_s^2}\,\alpha\, E_p\, e^{-\alpha z} \sum_{n=1}^{\infty} \left|\phi(x - n\Delta t_s(v_0 + \Omega z), y, z)\right|^2, \tag{1}$$

Manuscript submitted October 14, 1976.

where

z = the distance in the propagation direction measured from the aperture plane,

x = the distance in the wind direction measured from beam maximum intensity in the aperture plane,

γ = the ratio of atmospheric specific heats ($\approx$1.4),

c_s = the speed of sound in air ($\approx$340 m/s),

α = the absorption coefficient for the laser radiation,

Δt_s = the pulse spacing,

E_p = the energy of each laser pulse,

v_0 = the wind speed along the x direction perpendicular to the direction of propagation, and

Ω = the angular sluing rate of the beam about the y axis.

Finally ϕ is the normalized steady-state energy distribution of each pulse at the z plane:

$$\int_{-\infty}^{\infty} |\phi(x, y, z)|^2 \, dxdy = 1. \tag{2}$$

This density reduction $\Delta\rho$ changes the index of refraction from its ambient value n_0, where $n_0 \approx 1$, to

$$n^2 \approx n_0^2 + 3N\Delta\rho,$$

where N is the molecular refractivity of air ($\approx$0.154 cm^3/g). The distribution ϕ must then be calculated self-consistently from the propagation equation:

$$\left[2ik \frac{\partial}{\partial z} + \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + 3Nk^2\Delta\rho(|\phi|^2)\right]\phi = 0, \tag{3}$$

where $k = 2\pi/\lambda$ is the wavenumber of the laser radiation. It is assumed in SSPARAMA that at $z = 0$ the pulse train has a spherical phase front and a truncated intensity profile. For example, when truncated Gaussian pulses are propagated

$$\begin{aligned} \phi(x, y, 0) &= N_g \phi_g(x, y), \qquad & x^2 + y^2 \leqslant 2a^2, \\ &= 0, \qquad & x^2 + y^2 > 2a^2, \end{aligned} \tag{4}$$

where

$$\phi_g(x, y) = \frac{1}{a\sqrt{\pi}} e^{-[1+(ika^2/f)][(x^2+y^2)/a^2]/2} \tag{5}$$

and N_g is a normalization constant insuring that Eq. (2) is satisfied at $z = 0$. Two scale lengths, a and f, are defined in Eq. (5). The scale length f, the initial curvature of the phase front, defines the distance from the aperture to the focal plane. At a distance a from the aperture center the beam intensity falls to $1/e$ of its maximum value, and the beam is truncated at $1/e^2$ of maximum intensity.

Altogether eight variable physical quantities, a, f, k, α, E_p, Δt_s, v_0, and Ω appear in Eqs. (1) through (5). All variations will not however lead to a mathematically distinct problem. In SSPARAMA Eqs. (1) through (5) are scaled so that distinct propagation problems are defined in terms of five dimensionless parameters. The program is designed to accept either the set of data with dimensions or the dimensionless set, and both sets are printed out.

The scaling of Eqs. (1) through (5) is carried out via the coordinate transformations

$$\tilde{x} \equiv \frac{x}{a}, \qquad \tilde{y} \equiv \frac{y}{a}, \qquad \tilde{z} \equiv \frac{z}{f} \tag{6}$$

and the variable transformation

$$\tilde{\phi}(\tilde{x}, \tilde{y}, \tilde{z}) \equiv a\phi(x, y, z). \tag{7}$$

By multiplying Eq. (3) through by a^3, one can write the propagation equation in a form which identifies the five dimensionless parameters characterizing propagation in SSPARAMA:

$$\left\{2iN_k \frac{\partial}{\partial \tilde{z}} + \frac{\partial^2}{\partial \tilde{x}^2} + \frac{\partial^2}{\partial \tilde{y}^2} - N_k N_c e^{-N_\alpha z} \sum_{n=1}^{\infty} \left| \tilde{\phi}\left[\tilde{x} - \frac{2n}{N_o}(1 + N_s \tilde{z}), \tilde{y}, \tilde{z}\right]\right|^2 \right\} \tilde{\phi} = 0. \tag{8}$$

The five parameters, N_k, N_c, N_α, N_o, and N_s, are defined as

$$N_k = ka^2/f, \tag{9}$$

$$N_c = \frac{3Nk(\gamma - 1)\alpha f E_p}{c_s^2 a^2}, \tag{10}$$

$$N_\alpha = \alpha f, \tag{11}$$

$$N_o = \frac{2a}{v_0 \Delta t_s}, \tag{12}$$

and

$$N_s = \Omega f / v_0. \tag{13}$$

N_k is the Fresnel number of the free-propagation problem, and N_c, N_α, N_o, and N_s are coupling strength, absorption, overlap, and sluing parameters respectively. N_o was introduced by Wallace and Lilly [4] and called the pulses-per-flow-time parameter. It measures the number of preceding pulses which have heated the air across the beam aperture as the pulse under study begins to propagate. The solution to Eq. (8) is obtained subject to the energy normalization

$$\int |\tilde{\phi}(\tilde{x}, \tilde{y}, 0)|^2 \, d\tilde{x} \, d\tilde{y} = 1 \tag{14}$$

and the initial condition

$$\tilde{\phi}(\tilde{x}, \tilde{y}, 0) = |\tilde{\phi}| \, e^{-iN_k(\tilde{x}^2+\tilde{y}^2)/2}, \tag{15}$$

where $|\tilde{\phi}| = 0$ for $\tilde{x}^2 + \tilde{y}^2 > 2$.

Equations (8), (14), and (15) are numerically solved in SSPARAMA on a 64-by-64 grid in the $\tilde{x}\tilde{y}$ plane. Since one would like to use as much of the computational grid as possible to describe the variations in beam intensity, a scheme for adapting the coordinate grid to the propagation must be used. For example, as the beam propagates, the initial focusing causes the beam intensity pattern to decrease in size until the negative lensing effects of the heated atmosphere accumulate to thermally defocus it. Moreover, since the wind removes heated air from the path of the beam from left to right, a thermal gradient is established that deflects the beam from right to left. If the computational grid were not moved or changed in size as the beam intensity was calculated from aperture to focal plane, the intensity pattern would either be poorly sampled as it decreased in size or it would expand or deflect to reach the boundary of the grid and invalidate the calculation.

A technique for adapting the computational grid to local changes in the size or location of the beam intensity pattern has been developed by Herrmann and Bradley [5]. A slightly modified form of their technique has been incorporated into SSPARAMA and will be described in the next section of this report. In the third section the numerical procedures used in SSPARAMA will be described, and in the fourth section the code usage will be explained.

COORDINATE-SYSTEM ADAPTION

The dimensionless form of the propagation equation can be rewritten more compactly as

$$[2iN_k \partial_{\tilde{z}} + \partial_{\tilde{x}}^2 + \partial_{\tilde{y}}^2 + k^2 a^2 (n^2 - 1)] \tilde{\phi} = 0, \tag{16}$$

where $n^2 - 1$, the nonlinear index of refraction, depends on $\tilde{\phi}$ as given by Eq. (8). The $\tilde{x}\tilde{y}\tilde{z}$ coordinate system is normalized to the constant lengths a and f, and is fixed in space. In this system therefore the beam will lie symmetrically about the origin of the $\tilde{x}\tilde{y}$ plane only at $\tilde{z} = 0$ with an extent of order 1 (see, for example, Eq. (15)). When $z \neq 0$, a new set of xy coordinates is needed to maintain the two properties that the beam be centered about the xy coordinate origin and be of order 1 in extent. In general, one can relate the xy and $\tilde{x}\tilde{y}$ coordinates by a set of scale parameters D_1 and D_2 and a deflection parameter X, which are functions of $\tilde{z}$. Since one would like to solve Eq. (16) in a set of coordinates that adapt to changes in beam size and direction, the coordinate transformation must be related to these beam changes as determined by the linear and quadratic terms of the phase front. By analogy therefore with the transformation to dimensionless parameters, one must perform simultaneous coordinate and variable transformations. The form of these transformations is suggested by linear propagation theory:

$$x = \frac{\tilde{x} - X}{\sqrt{D_1}}, \tag{17}$$

$$y = \frac{\tilde{y}}{\sqrt{D_2}}, \tag{18}$$

$$z = \frac{\tilde{z}}{N_k}, \tag{19}$$

and

$$\tilde{\phi} = \frac{\psi}{\sqrt[4]{D_1 D_2}} \, e^{i(\tilde{\alpha}_1 \tilde{x}^2 + \tilde{\alpha}_2 \tilde{y}^2 + \tilde{\beta}\tilde{x} + \tilde{\gamma}_1 + \tilde{\gamma}_2)}. \tag{20}$$

The constant scale change from $\tilde{z}$ to z is done for convenience to eliminate N_k from the z-derivative term in Eq. (16):

$$2iN_k \partial_{\tilde{z}} \rightarrow 2i\partial_z .$$

The factor $1/\sqrt[4]{D_1 D_2}$ is removed from $\tilde{\phi}$ to insure the form invariance of the energy normalization:

$$\int |\tilde{\phi}|^2 \, d\tilde{x}d\tilde{y} = \int |\psi|^2 \, dxdy = 1. \tag{21}$$

When Eqs. (17) through (20) are substituted into Eq. (16) and when the nonlinear term is of negligible size and the beam has a Gaussian profile, D_1, D_2, X, $\tilde{\alpha}_1$, $\tilde{\alpha}_2$, $\tilde{\beta}$, $\tilde{\gamma}_1$, and $\tilde{\gamma}_2$ as functions of z can be analytically determined for all z. However, when the nonlinear term is important or when a non-Gaussian beam is propagated, the $\tilde{\alpha}$'s and $\tilde{\beta}$, which represent the effective quadratic and linear phase changes throughout the xy plane, can no longer be so determined. One must adopt a more limited strategy for the employment of Eqs. (17) through (20).

Consider, for example, that the quantities D_1, D_2, X, $\tilde{\alpha}_1$, $\tilde{\alpha}_2$, $\tilde{\beta}$, $\tilde{\gamma}_1$, and $\tilde{\gamma}_2$ are known at $z = z_0$ and that their dependence on z is to be analytically determined as one propagates to a neighboring xy plane at $z_0 + \Delta z$. Since

$$\partial_{\tilde{x}}^2 = \frac{1}{D_1} \partial_x^2, \tag{22}$$

$$\partial_{\tilde{y}}^2 = \frac{1}{D_2} \partial_y^2, \tag{23}$$

and

$$\partial_{\tilde{z}} = \frac{1}{N_k}\left[\partial_z - \left(\frac{x}{2}\partial_z \ln D_1 + \frac{\partial_z X}{\sqrt{D_1}}\right)\partial_x - \frac{y}{2}\partial_z \ln D_2 \,\partial_y\right], \tag{24}$$

one finds that

$$[2iN_k\partial_{\tilde{z}} + \partial_{\tilde{x}}^2 + \partial_{\tilde{y}}^2 + k^2a^2(n^2-1)]\frac{\psi}{\sqrt[4]{D_1D_2}}\; e^{i(\tilde{\alpha}_1\tilde{x}^2+\tilde{\alpha}_2\tilde{y}^2+\tilde{\beta}\tilde{x}+\tilde{\gamma}_1+\tilde{\gamma}_2)}$$

$$= \frac{e^{i(\tilde{\alpha}_1\tilde{x}^2+\tilde{\alpha}_2\tilde{y}^2+\tilde{\beta}\tilde{x}+\tilde{\gamma}_1+\tilde{\gamma}_2)}}{\sqrt[4]{D_1D_2}}\left\{2i\left(\partial_z - \frac{x}{2}\partial_z \ln D_1\,\partial_x - \frac{1}{\sqrt{D_1}}\partial_z X\partial_x - \frac{y}{2}\partial_z \ln D_2\,\partial_y\right)\right.$$

$$- \frac{i}{2}(\partial_z \ln D_1 + \partial_z \ln D_2) - 2\partial_z(\tilde{\gamma}_1+\tilde{\gamma}_2) + \frac{1}{D_1}\partial_x^2 - [2\tilde{\alpha}_1(\sqrt{D_1}\,x + X) + \tilde{\beta}]^2$$

$$+ \frac{2i}{\sqrt{D_1}}[2\tilde{\alpha}_1(\sqrt{D_1}\,x + X) + \tilde{\beta}]\partial_x + 2i\tilde{\alpha}_1 + \frac{1}{D_2}\partial_y^2 - 4\tilde{\alpha}_2^2D_2y^2$$

$$\left. + 4i\tilde{\alpha}_2 y\partial_y + 2i\tilde{\alpha}_2 + k^2a^2(n^2-1)\right\}\psi = 0. \tag{25}$$

For vanishingly small $n^2 - 1$ and for a real Gaussian profile $\psi(x, y, z_0)$ one would determine D_1, D_2, X, $\tilde{\alpha}_1$, $\tilde{\alpha}_2$, $\tilde{\beta}$, $\tilde{\gamma}_1$, and $\tilde{\gamma}_2$ from the requirement that Eq. (25) be capable of being put in the form

$$\left[2i\,\partial_z + \frac{1}{D_1}(\partial_x^2 + 1 - x^2) + \frac{1}{D_2}(\partial_y^2 + 1 - y^2) + k^2a^2(n^2-1)\right]\psi = 0. \tag{26}$$

Then, as ψ was propagated to $z_0 + \Delta z$, it would acquire no z dependence and would remain real and Gaussian; that is, all of the z dependence of ϕ would have been accounted for in $D_1, \cdots, \tilde{\gamma}_2$.

For the imaginary terms of Eq. (25) other than $2i\,\partial_z$ to vanish, the quantities D_1, D_2, and X, which determine the scale and location of the xyz coordinate system, must satisfy the equations

$$\partial_z \ln D_1 = 4\tilde{\alpha}_1, \tag{27}$$

$$\partial_z \ln D_2 = 4\tilde{\alpha}_2, \tag{28}$$

and

$$\partial_z X = 2\tilde{\alpha}_1 X + \tilde{\beta}. \tag{29}$$

On the other hand, for the real terms involving ∂_x and ∂_y to vanish and for the scale functions D_1 and D_2 to be factorable from the remaining x and y terms respectively, the phase functions $\tilde{\alpha}_1$, $\tilde{\alpha}_2$, $\tilde{\beta}$, $\tilde{\gamma}_1$, and $\tilde{\gamma}_2$ must satisfy the set of equations

$$2D_1\partial_z\tilde{\alpha}_1 + 4\tilde{\alpha}_1^2 D_1 = \frac{1}{D_1}, \tag{30}$$

$$2D_2\partial_z\tilde{\alpha}_2 + 4\tilde{\alpha}_2^2 D_2 = \frac{1}{D_2}, \tag{31}$$

$$\partial_z\tilde{\beta} + 2X\partial_z\tilde{\alpha}_1 + 2\tilde{\alpha}_1(2\tilde{\alpha}_1 X + \tilde{\beta}) = 0, \tag{32}$$

$$2\partial_z\tilde{\gamma}_1 + 2X^2\partial_z\tilde{\alpha}_1 + 2X\partial_z\tilde{\beta} + (2\tilde{\alpha}_1 X + \tilde{\beta})^2 = -\frac{1}{D_1}, \tag{33}$$

and

$$2\partial_z\tilde{\gamma}_2 = -\frac{1}{D_2}. \tag{34}$$

Thus Eqs. (27) through (34) will determine all of the z dependence of $\tilde{\phi}$ when $\psi(x, y, z_0)$ is real and a Gaussian function of x and y and there is no lensing effect caused by heating of the atmosphere; that is, Eqs. (27) through (34) will describe beam focusing in the absence of diffraction and nonlinear media phenomena. They are of more limited utility when such phenomena are present. In this case, during the displacement of ϕ from z_0 to $z_0 + \Delta z$, linear and quadratic phase changes will arise from two sources. As a result of focusing at $z = z_0$, the initial phases $\tilde{\alpha}_1(z_0)$, $\tilde{\alpha}_2(z_0)$, and $\tilde{\beta}(z_0)$ will become $\tilde{\alpha}_1(z_0 + \Delta z)$, $\tilde{\alpha}_2(z_0 + \Delta z)$, and $\tilde{\beta}(z_0 + \Delta z)$ through the solution to Eqs. (27) through (34). In addition however ψ at $z_0 + \Delta z$ will acquire linear and quadratic phases, $\Delta\tilde{\beta}$, $\Delta\tilde{\alpha}_1$, and $\Delta\tilde{\alpha}_2$ respectively, as a result of diffraction and thermal blooming. Thus at $z_0 + \Delta z$ a new factorization of $\tilde{\phi}$ must be made, namely,

$$\tilde{\phi}(\tilde{x}, \tilde{y}, \tilde{z}_0 + \Delta\tilde{z}) \equiv \frac{\psi'(x, y, z_0 + \Delta z)}{\sqrt[4]{D_1(z_0 + \Delta z)D_2(z_0 + \Delta z)}}\, e^{i[\tilde{\alpha}_1'(z_0+\Delta z)\tilde{x}^2 + \tilde{\alpha}_2'(z_0+\Delta z)\tilde{y}^2 + \tilde{\beta}'(z_0+\Delta z)\tilde{x} + \tilde{\gamma}_1' + \tilde{\gamma}_2']}, \tag{35}$$

if ψ', which is to be propagated from $z_0 + \Delta z$ to $z_0 + \Delta z + \Delta z'$, is not to initially have quadratic or linear phase terms. After each step in propagation therefore $\tilde{\alpha}_1$, $\tilde{\alpha}_2$, and $\tilde{\beta}$ must be redefined as

$$\tilde{\alpha}_1'(z_0 + \Delta z) = \tilde{\alpha}_1(z_0 + \Delta z) + \Delta\tilde{\alpha}_1, \tag{36}$$

$$\tilde{\alpha}_2'(z_0 + \Delta z) = \tilde{\alpha}_2(z_0 + \Delta z) + \Delta\tilde{\alpha}_2, \tag{37}$$

and

$$\tilde{\beta}'(z_0 + \Delta z) = \tilde{\beta}(z_0 + \Delta z) + \Delta\tilde{\beta} \tag{38}$$

in order to adapt the coordinate-system determination from Eqs. (27) through (29) to changes in phase that result from focusing, diffraction, and thermal blooming.

In SSPARAMA, ψ is propagated from one z plane to another by finite-differencing a phase-transformed version of Eq. (26). Then $\Delta\alpha_1$, $\Delta\alpha_2$, and $\Delta\beta$ are *found in the xyz coordinate system* using the method of phase minimization discussed by Herrmann and Bradley [5]. One requires that

$$\int_{z=z_0+\Delta z} |\psi|^2[\nabla(\Delta\alpha_1 x^2 + \Delta\alpha_2 y^2 + \Delta\beta x - \gamma)]^2\, dxdy = \text{minimum}, \tag{39}$$

where $\psi(x, y, z_0 + \Delta z) \equiv |\psi| e^{i\gamma}$. It follows that

$$\Delta\alpha_1 = \frac{D_1 E - B_1 C_1}{2(A_1 E - B_1^2)}, \tag{40}$$

$$\Delta\beta = \frac{A_1 C_1 - B_1 D_1}{A_1 E - B_1^2}, \tag{41}$$

and

$$\Delta\alpha_2 = \frac{D_2}{2A_2}, \tag{42}$$

where

$$A_1 \equiv \int x^2 |\psi|^2\, dxdy, \quad A_2 \equiv \int y^2 |\psi|^2\, dxdy, \tag{43}$$

$$B_1 \equiv \int x|\psi|^2\, dxdy, \tag{44}$$

$$C_1 \equiv \text{Im} \int \psi^* \partial_x \psi\, dxdy, \tag{45}$$

$$D_1 \equiv \text{Im} \int x\psi^* \partial_x \psi\, dxdy, \quad D_2 \equiv \text{Im} \int y\psi^* \partial_y \psi\, dxdy, \tag{46}$$

and

$$E \equiv \int |\psi|^2\, dxdy = 1. \tag{47}$$

The factorization

$$\psi(x, y, z_0 + \Delta z) \equiv \psi' e^{i(\Delta\alpha_1 x^2 + \Delta\alpha_2 y^2 + \Delta\beta x)} \tag{48}$$

will then define ψ' at $z_0 + \Delta z$ as a wave function of minimum quadratic and linear phase. In particular, if ψ is exactly a Gaussian beam, ψ' will be real.

The relationship between $\{\Delta\alpha_1, \Delta\alpha_2, \Delta\beta\}$ and $\{\Delta\tilde{\alpha}_1, \Delta\tilde{\alpha}_2, \Delta\tilde{\beta}\}$ is found by substituting Eqs. (17) and (18) into Eq. (48):

$$\Delta\tilde{\alpha}_1 = \frac{\Delta\alpha_1}{D_1}, \tag{49}$$

$$\Delta\tilde{\alpha}_2 = \frac{\Delta\alpha_2}{D_2}, \tag{50}$$

and

$$\Delta\tilde{\beta} = \frac{\Delta\beta}{\sqrt{D_1}} - \frac{2\Delta\alpha_1 X}{D_1}. \tag{51}$$

A similar set of equations will hold between $\{\tilde{\alpha}_1, \tilde{\alpha}_2, \tilde{\beta}\}$ and $\{\alpha_1, \alpha_2, \beta\}$, which are computed directly in the *xyz* coordinate system. When reexpressed in terms of α_1, α_2 and β, Eqs. (27) through (29) become

$$\partial_z D_1 = 4\alpha_1, \tag{52}$$

$$\partial_z D_2 = 4\alpha_2, \tag{53}$$

and

$$\partial_z X = \frac{\beta}{\sqrt{D_1}}, \tag{54}$$

and Eqs. (30) through (32) transform into

$$\partial_z \alpha_1 = \frac{1}{2D_1}(1 + 4\alpha_1^2), \tag{55}$$

$$\partial_z \alpha_2 = \frac{1}{2D_2}(1 + 4\alpha_2^2), \tag{56}$$

and

$$\partial_z \beta = \frac{2\alpha_1\beta}{D_1}. \tag{57}$$

Eqs. (52) through (57) must be solved in terms of initial values at z_0. The solutions are

$$D_{1,2}(z) = D_{1,2}(z_0)\left\{\left[1 + \frac{2\alpha_{1,2}(z_0)}{D_{1,2}(z_0)}(z - z_0)\right]^2 + \left[\frac{z - z_0}{D_{1,2}(z_0)}\right]^2\right\}, \tag{58}$$

$$\alpha_{1,2}(z) = \alpha_{1,2}(z_0) + \frac{1}{2}\{1 + [2\alpha_{1,2}(z_0)]^2\}\frac{z - z_0}{D_{1,2}(z_0)}, \tag{59}$$

$$\beta(z) = \beta(z_0)\sqrt{\left[1 + \frac{2\alpha_1(z_0)}{D_1(z_0)}(z - z_0)\right]^2 + \left[\frac{z - z_0}{D_1(z_0)}\right]^2}, \tag{60}$$

and

$$X(z) = X(z_0) + \frac{\beta(z_0)}{\sqrt{D_1(z_0)}}(z - z_0). \tag{61}$$

Finally the procedure for solving Eq. (26) in SSPARAMA is similar to the one described in an earlier report [2]. A phase transformation on ψ is made:

$$\Phi(x, y, z) \equiv \psi(x, y, z)\, e^{-(i/2)\int_{z_0}^{z} g(x,y,z')dz'}, \tag{62}$$

where

$$g(x, y, z) \equiv \frac{1}{D_1}(1 - x^2) + \frac{1}{D_2}(1 - y^2) + k^2a^2(n^2 - 1). \tag{63}$$

The equation for Φ follows from Eq. (26):

$$[2i\,\partial_z + H(x, y, z)]\Phi = 0, \tag{64}$$

where

$$H = e^{-(i/2)\int_{z_0'}^{z} g\,dz'}\left(\frac{1}{D_1}\partial_x^{\,2} + \frac{1}{D_2}\partial_y^{\,2}\right)e^{(i/2)\int_{z_0}^{z} g\,dz'}. \tag{65}$$

By picking z_0' to lie between z_0 and $z_0 + \Delta z$, one can propagate Φ from z_0 to $z_0 + \Delta z_0$, with first-order accuracy, by solving the equation

$$[2i\partial_z + H(x, y, z_0')]\Phi = \left(2i\partial_z + \frac{1}{D_1}\partial_x^{\,2} + \frac{1}{D_2}\partial_y^{\,2}\right)\Phi = 0. \tag{66}$$

Equation (66) is solved by Fourier transforming Φ [6],

$$\tilde{\Phi}(k_1, k_2, z_0) \equiv \int e^{i(k_1x+k_2y)}\,\Phi(x, y, z_0)\,dxdy, \tag{67}$$

and propagating $\tilde{\Phi}$ to $z_0 + \Delta z$:

$$\tilde{\Phi}(k_1, k_2, z_0 + \Delta z) = \tilde{\Phi}(k_1, k_2, z_0)\,e^{(i/2)\{k_1^2\int_{z_0}^{z_0+\Delta z}[1/D_1(z)]dz + k_2^2\int_{z_0}^{z_0+\Delta z}[1/D_2(z)]dz\}}. \tag{68}$$

The inverse transformation to Eq. (67) then yields Φ, and Eq. (62) yields $\psi(x, y, z_0 + \Delta z)$.

NUMERICAL PROCEDURES

The phase function $g(x, y, z)$ of Eq. (63) can be written more usefully in the form

$$g = \frac{g_1(x)}{D_1(z)} + \frac{g_2(y)}{D_2(z)} - \frac{g_3(x, y, z)}{\sqrt{D_1(z)D_2(z)}}, \tag{69}$$

where

$$g_1(x) \equiv 1 - x^2, \tag{70}$$

$$g_2(y) \equiv 1 - y^2, \tag{71}$$

and

$$g_3(x, y, z) \equiv N_k N_c e^{-N_\alpha N_k z} \sum_{n=1}^{\infty} \left| \Phi \left[x - \frac{2n}{N_0\sqrt{D_1(z)}}(1 + N_s N_k z), y, z \right] \right|^2. \tag{72}$$

This expression for g_3 is found by substituting the new variables x, y, z, and Φ into Eq. (8). The phase integral

$$\Delta\theta \equiv \int_{z_0'}^{z} g(x, y, z')\, dz'$$

appearing in Eq. (62) can now be partially evaluated and expressed in the form

$$\Delta\theta = g_1(x)\Delta Z_1 + g_2(y)\Delta Z_2 - \int_{z_0'}^{z} \frac{g_3(x, y, z)}{\sqrt{D_1(z)D_2(z)}}\, dz, \tag{73}$$

where

$$\Delta Z_{1,2} \equiv \int_{z_0'}^{z} \frac{dz'}{D_{1,2}(z')} = \tanh^{-1} \left(\{1 + [2\alpha_{1,2}(z_0)]^2\} \frac{z' - z_0}{D_{1,2}(z_0)} + 2\alpha_{1,2}(z_0) \right) \Bigg|_{z'=z_0'}^{z'=z}. \tag{74}$$

The differential quantities ΔZ_1 and ΔZ_2 are similarly named as the coordinate differential ΔZ that was used in earlier code calculations which involved only a single scaling function $D(z)$.

To complete the evaluation of $\Delta\theta$, one must know the z dependence of g_3, that is, the z dependence of $|\Phi|^2$. Two options are provided in SSPARAMA, for evaluating $\Delta\theta$, depending on whether one has determined $|\Phi|^2$ at one or both of the integration endpoints. The procedures work as follows: Suppose first that the solution for $\psi(x, y, z_0)$ has been obtained. Then one can compute $g(x, y, z_0)$, since $|\Phi(x, y, z_0)|^2 = |\psi(x, y, z_0)|^2$. To find $\Phi(x, y, z_0)$, however, one must evaluate

$$\Delta\theta' \equiv \int_{z_0}^{z_0'} g(x, y, z')\, dz', \tag{75}$$

where z_0' lies between z_0 and the plane $z_0 + \Delta z$ to which one would like to propagate ψ. If ψ is known only at z_0, the zeroth-order approximation

$$\Delta\theta' \approx g_1(x)\Delta Z_1' + g_2(y)\Delta Z_2' - g_3(x, y, z_0)\Delta Z_{12}' \tag{76}$$

must be made, where

$$\Delta Z_{12}' \equiv \int_{z_0}^{z_0'} \frac{dz'}{\sqrt{D_1(z')D_2(z')}}. \tag{77}$$

Equation (66) can now be solved for $\Phi(x, y, z_0 + \Delta z)$ by the use of Fourier transformations. Finally on performance of the phase integral

$$\Delta\theta'' \equiv \int_{z_0'}^{z_0+\Delta z} g(x, y, z')\, dz' \tag{78}$$

$\psi(x, y, z_0 + \Delta z)$ can be obtained from $\Phi(x, y, z_0 + \Delta z)$. In keeping with the accuracy with which $\Delta\theta'$ was approximated, $\Delta\theta''$ can be approximately evaluated as

$$\Delta\theta'' \approx g_1(x)\Delta Z_1'' + g_2(y)\Delta Z_2'' - g_3(x, y, z_0 + \Delta z)\Delta Z_{12}''. \tag{79}$$

The differentials $\Delta Z_1''$, $\Delta Z_2''$, and $\Delta Z_{12}''$ are defined by the integrals of Eqs. (74) and (77) with the integration limits as specified in Eq. (78).

Suppose however that initially both $\psi(x, y, z_0)$ and $\psi(x, y, z_0')$ are known and that the values of ψ at z_0 are to be propagated to the plane at $z_0 + \Delta z$. In this case the phase integrals defined in Eqs. (75) and (78) can be approximated using the integration formula

$$\int_{x_0}^{x_0+\Delta x} f(x)g(x)\, dx \approx w_1 f(x_0) + w_2 f(x_0 + \Delta x), \tag{80}$$

which has first-order instead of zeroth-order accuracy. The weights w_1 and w_2 are thus determined such that equality will hold in Eq. (80) whenever f is a linear function of x:

$$w_1 = \left(1 + \frac{2x_0}{\Delta x}\right)\int_{x_0}^{x_0+\Delta x} g(x)\, dx - \frac{2}{\Delta x}\int_{x_0}^{x_0+\Delta x} xg(x)\, dx \tag{81}$$

and

$$w_2 = \frac{2}{\Delta x}\int_{x_0}^{x_0+\Delta x} xg(x)\,dx - \frac{2x_0}{\Delta x}\int_{x_0}^{x_0+\Delta x} g(x)\,dx. \tag{82}$$

Then, for example, in place of Eq. (76) one would have that

$$\Delta\theta' \approx g_1(x)\Delta Z_1' + g_2(y)\Delta Z_2' - g_3(x, y, z_0)\Delta Z_3' - g_3(x, y, z_0')\Delta Z_4', \tag{83}$$

where $\Delta Z_3'$ and $\Delta Z_4'$ are related through Eqs. (81) and (82) to $\Delta Z_{12}'$ and an integration over the function $z/\sqrt{D_1(z)D_2(z)}$:

$$\Delta Z_3' = \frac{z_0' + z_0}{z_0' - z_0}\Delta Z_{12}' - \frac{2}{z_0' - z_0}\int_{z_0}^{z_0'} \frac{z'\,dz'}{\sqrt{D_1(z')D_2(z')}} \tag{84}$$

and

$$\Delta Z_4' = \frac{2}{z_0' - z_0}\left[\int_{z_0}^{z_0'} \frac{z'\,dz'}{\sqrt{D_1(z')D_2(z')}} - z_0\Delta Z_{12}'\right]. \tag{85}$$

Although integrations over D_1^{-1} and D_2^{-1} can be carried out analytically in terms of inverse hyperbolic tangents (as in Eq. (74)), integrals over $1/\sqrt{D_1D_2}$ produce elliptic functions. Both sets of integrations are handled in SSPARAMA numerically, with third-order accuracy, using a second integration formula:

$$\int_{x_0}^{x_0+\Delta x} f(x)\,dx \approx \frac{\Delta x}{2}\left[f(x_0 + \Delta x_1) + f(x_0 + \Delta x_2)\right], \tag{86}$$

where $\Delta x_1 \equiv (1 - 1/\sqrt{3})\Delta x/2$ and $\Delta x_2 \equiv (1 + 1/\sqrt{3})\Delta x/2$. Again, as an example, consider Eqs. (84) and (85) and define

$$f_1 \equiv \frac{1}{\sqrt{D_1(z_1)D_2(z_1)}} \tag{87}$$

and

$$f_2 \equiv \frac{1}{\sqrt{D_1(z_2)D_2(z_2)}}, \tag{88}$$

where $z_1 \equiv z_0 + (1 - 1/\sqrt{3})[(z_0' - z_0)/2]$ and $z_2 \equiv z_0 + (1 + 1/\sqrt{3})[(z_0' - z_0)/2]$. One can complete the numerical evaluation of $\Delta Z_3'$ and $\Delta Z_4'$ by rewriting Eqs. (84) and (85) with the use of Eq. (86), in terms of f_1 and f_2:

$$\Delta Z_3' = \frac{z_0' - z_0}{2\sqrt{3}}(f_1 - f_2) \tag{89}$$

and

$$\Delta Z_4' = \frac{z_0' - z_0}{2}\left[\left(1 - \frac{1}{\sqrt{3}}\right)f_1 + \left(1 + \frac{1}{\sqrt{3}}\right)f_2\right]$$

$$= (z_0' - z_0)\left(\frac{f_1 + f_2}{2}\right) - \Delta Z_3'. \tag{90}$$

The procedure by which Eqs. (80) through (90) are employed requires that two sets of values of ψ be stored at any time by SSPARAMA. At the beginning of the propagation step described above, the two arrays contain the values of $\psi(x, y, z_0)$ and $\psi(x, y, z_0')$, where $z_0 < z_0' < z_0 + \Delta z$. At the end of the propagation step the values of $\psi(x, y, z_0)$ have been replaced by $\psi(x, y, z_0 + \Delta z)$. These new values can then be used to propagate $\psi(x, y, z_0')$ to $\psi(x, y, z_0' + \Delta z')$, where now $z_0' < z_0 + \Delta z < z_0' + \Delta z'$. The process of alternatively propagating one and then the other of the two arrays is repeated until the focal plane, defined by the initial beam curvature, is reached.

Since both arrays are initially assigned the values $\psi(x, y, 0)$, the process of propagating one array past the other cannot begin until after the first propagation step. The first z step is therefore taken using Eqs. (76) and (79) to determine $\Delta\theta'$ and $\Delta\theta''$. In general the incremental steps Δz are selected in SSPARAMA according to a criterion that the phase changes induced by g_3 as computed from Eq. (76) be no larger than some preassigned value of order 1 for all x and y. However, to carry out the first advancement of ψ at $z_0 = 0$, half of the initially computed Δz value is used. This leapfrog procedure is summarized for the first few z steps in Fig. 1.

The advantage conveyed by using Eqs. (76) and (79) to evaluate the phase integrals $\Delta\theta'$ and $\Delta\theta''$ is that only one ψ array is needed in carrying out the calculation. Because of the reduced accuracy in computing $\Delta\theta'$ and $\Delta\theta''$, however, smaller z steps are in principle required to obtain the same results as when two arrays at different z planes are used. To allow a quantitative comparison of these two procedures, both options for propagating ψ were installed in SSPARAMA and can be selected according to the value of one of the input parameters to the code. For the same reason, another input parameter is also available that allows one to adapt or not adapt the coordinate system to the amount of diffraction or thermal blooming occurring during beam propagation.

PROGRAM OPERATION

This section will describe the input parameters required to run SSPARAMA and explain the data included in the output. A complete listing of SSPARAMA is included in Appendix A.

To use program SSPARAMA, two input cards are required. The first specifies certain numerical parameters and selects various program options, and the second defines the

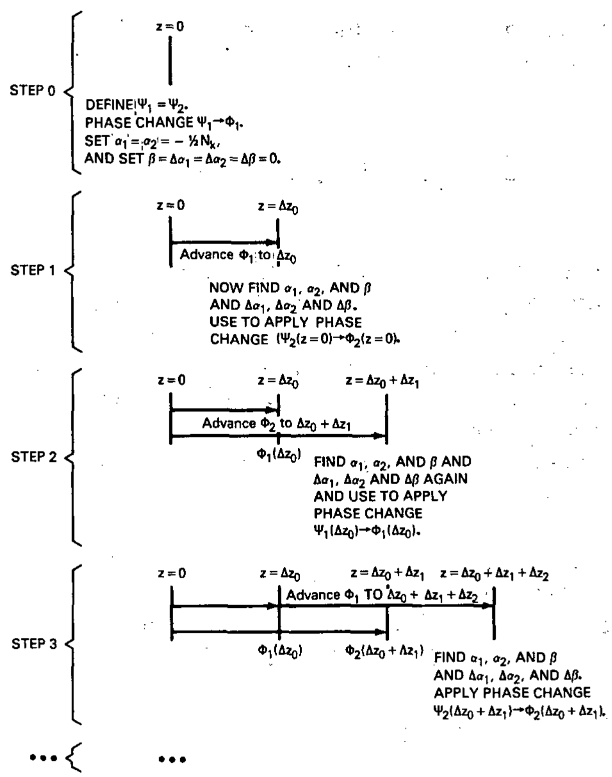

Fig. 1—Leapfrog procedure for advancing the wave function Φ

particular physical situation. This second card can contain the actual physical parameters or a set of dimensionless parameters.

First Input Card

The parameters read from the first card are listed in Table 1. A description of each of these parameters is as follows:

Table 1—Parameters Specified by the First Input Card

Columns	Name	Format	Columns	Name	Format
1-5	PHIMXX	F5.0	36-40	NPM	I5
6-10	ROCULT	F5.0	41-45	NBM	I5
11-15	HXY	F5.0	46-50	NPLOT	I5
16-20	NXY	I5	51-55	NCT	I5
21-25	NCW	I5	56-60	NRS	I5
26-30	NAD	I5	61-65	NPUNCH	I5
31-35	NMS	I5	75-80	NID	A6

PHIMXX. This is the maximum allowed phase change in radians for any point in the computational grid at each z step. It is used to define the newly computed z increments HZN at each step, where

$$\frac{g_3(x, y, z)_{\max}}{\sqrt{D_1 D_2}} \text{ HZN} = \text{PHIMXX},$$

in which $g_3(x, y, z)_{\max}$ is the maximum value in the computational grid of g_3, given by Eq. (72). PHIMXX is nominally entered as 1.0. If more z steps are required, PHIMXX can be decreased. In this case the z increment is tied to the amount of heating in the atmosphere, becoming smaller automatically as large density changes take place or becoming large and efficient when near-vacuumlike propagation occurs. If HZN exceeds 0.1 of the total propagation distance, the smaller of these two z increments is used. If HZN at any time is less than 10^{-7} times the distance to be propagated, the program exits and an error message will be printed.

ROCULT. This is used when propagating uniform circular beamshapes with an obscuring disk or a uniform rectangular beamshape. In the former case ROCULT is the ratio of the occulting radius to the total radius. For a rectangle, it is the ratio of the y to the x dimension. ROCULT is used only when NBM equals 4 or 5.

HXY. This parameter defines the size of the computational grid relative to the aperture radius by

$$\Delta x = \Delta y = \text{HXY}$$

where Δx and Δy are the sizes of individual computational cells, which start out square. Depending on the beamshape, values between 0.1 and 0.3 are typical.

NXY. This is the number of individual computational cells along the edge of the entire computational grid. The FFT routine is more efficient when NXY is a power of 2, and NXY is normally entered as 64.

NCW. This parameter permits CW propagation to be included by allowing the summation in Eq. (72) to be replaced by an integral [7]. Before the summation is replaced, Eq. (72) can be written in terms of physical parameters as

$$\frac{3N(\gamma - 1)k^2 \alpha E_p e^{-\alpha z}}{c_s^2} \sum_{n=1}^{\infty} |\Phi[x - n(v_0 + \Omega z)\Delta t, y, z]|^2 .$$

This summation is performed when NCW = 0. When NCW = 1, the program is in the CW mode, and Eq. (72) is replaced by

$$\frac{3N(\gamma - 1)k^2 \alpha P e^{-\alpha z} \sqrt{D_1}}{c_s^2 (v_0 + \Omega z)} \int_{-\infty}^{0} |\Phi(x + x', y, z)|^2 \, dx',$$

where P is the average power of a CW laser ($P = E_p/\Delta t$). The integration is performed using a simple trapezoid rule.

NAD. When NAD = 0, the coordinate system adaption is not included. When NAD = 1, it is included.

NMS. When NMS = 0, the midplane integrations are not used. When NMS = 1, they are used.

NPM. When NPM = −1, the second data card contains physical parameters. When NPM = +1, the second card contains dimensionless parameters.

NBM. This parameter selects one of the five beamshapes available within the program:

NBM = 0 — Infinite Gaussian, with WIDTH (a parameter read from the second input card) being the e^{-1} intensity radius;

NBM = 1 — Truncated Gaussian, with WIDTH being the e^{-1} intensity radius, truncated at $\sqrt{2} \times$ WIDTH or e^{-2} intensity radius;

NBM = 2 — Uniform circular aperture, with WIDTH being the actual aperture radius;

NBM = 3 — Uniform square aperture, with WIDTH being the dimension from the center of the square to the edge (half-side dimension) in the x or y direction;

NBM = 4 — Uniform circular aperture and an occulting disk, with WIDTH being the total aperture radius and, as stated previously, with ROCULT being the ratio giving the occulting disk radius;

NBM = 5 — Uniform rectangular aperture, with WIDTH being the half-side x dimension and ROCULT being the ratio giving the y dimension.

NPLOT. This determines the type and the number of plots given in the output:

NPLOT = 0 — No plots;

NPLOT = 1 — Final contour plot only;

NPLOT = 2 — Final contour plot plus a plot of average intensity and peak intensity versus z;

NPLOT = 3 — Preceding plots plus a plot of flux and area versus irradiance;

NPLOT = 4 — Preceding plots plus a contour plot of aperture intensity;

NPLOT = 5 — Preceding plots plus Fourier-transform contour plots of aperture and final intensity distributions.

NCT. This determines the contour levels used in the contour plots:

NCT = 0 — Contour plots use contour levels with 10% increments;

NCT = 1 — Contour plots use 3-dB contours (0.5^n, $n = 1, 2, \ldots, 10$).

NRS. When NRS = 1, the final contour plot is corrected and standardized according to an internal criterion, to remove the effects of different amounts of coordinate system adaption in the x and y directions. When NRS = 0, this plot can appear with nonuniform axes.

NPUNCH. This determines whether there is a punched-card output:

NPUNCH = 0 — No punched-card output;

NPUNCH = 1 — Punched-card output for later data processing.

NID. Up to six characters can be used to identify a run or a series of runs on both the printed and punched output.

Second Input Card

The data contained on the second input card depend on the value of NPM. If NPM = −1, the physical parameters listed in Table 2 will be read. A description of each of these parameters is as follows:

Table 2—Parameters Specified by the Second Input Card When NPM = −1

Columns	Name	Format
1-5	OM	F5.0
6-10	HT	F5.0
11-15	ALPHA	F5.0
16-20	ALPHAS	F5.0
21-30	WIDTH	E10.0
31-40	WN	E10.0
41-50	VO	E10.0
51-60	ENERGY	E10.0
61-70	F	E10.0
71-80	ZF	E10.0

OM. The slew rate in radians per second.

HT. The interval between pulses in seconds, or the reciprocal of the pulse repetition frequency (PRF). For CW propagation this should be set to 1 second.

ALPHA. The absorption coefficient α in km^{-1}.

ALPHAS. The scattering coefficient in km^{-1}. ALPHAS is used to compute the total extinction but is not included in the absorption that produces atmospheric heating.

WIDTH. The aperture radius a in centimeters. The particular definition is given in the preceding subsection for each value of NBM.

WN. The wavenumber $k = 2\pi/\lambda$ or $2\pi/\beta\lambda$, where β is the beam quality and λ is the beam wavelength in centimeters.

VO. The wind velocity v_0 in meters per second.

ENERGY. The individual pulse energy E_p in joules. For CW propagation ENERGY is the average power in watts.

F. The focal length in kilometers.

ZF. The distance at which the calculation is to be stopped in kilometers.

As already shown, the propagation is a function of five dimensionless parameters. Different combinations of the eight physical parameters, which are required to define these dimensionless parameters and which lead to the same values of the dimensionless parameters, will produce identical results. In order that a unique physical situation be specified, some physical quantities are also read from the second data card when NPM = +1 (Table 3). They are not used to define the physical situation but rather to assign units to the derived quantities at the end of the calculations. The quantities read when NPM = +1 are:

Table 3—Quantities Specified by the Second Input Card When NPM = +1

Columns	Name	Format
1-5	F	F5.0
6-10	HT	F5.0
11-20	PNA	E10.0
21-30	PNALF	E10.0
31-40	PNK	E10.0
41-50	PNO	E10.0
51-60	PNS	E10.0
61-70	PND	E10.0
71-80	PNZ	E10.0

F. Focal length in kilometers.

HT. Pulse interval Δt in seconds (=1 second for CW).

PNA. The f number = WIDTH/F.

PNALF. Absorption number, ALPHA/F.

PNK. Fresnel number, $\mathrm{WN} \cdot \mathrm{WIDTH}^2/\mathrm{F}$.

PNO. Overlap number, $2\sqrt{2} \cdot \mathrm{WIDTH}/(\mathrm{VO} \cdot \mathrm{HT})$ for an infinite and truncated Gaussian beam and $2 \cdot \mathrm{WIDTH}/(\mathrm{VO} \cdot \mathrm{HT})$ for all other beam shapes.

PNS. Slew number, OM • F/VO.

PND. Distortion number, $3Nk(\gamma-1)\alpha f E_p/c_s^2 a\, v_0 \Delta t$.

PNZ. The ratio of the distance at which the calculation is to be stopped to the focal length, ZF/F.

Examples of Output

A series of multipulse runs was made varying the pulse spacing and energy so that the average power remained constant and using a number of average powers. The results of these runs are shown in Fig. 2 in the form of power optimization curves. The CW curve is included so that the convergence of the multipulse curves to the CW curve, as the limiting case when pulse interval is decreased, can be readily observed.

To test the SSPARAMA code in the CW mode, some comparison runs were made to check against some results obtained from Jan Herrmann of Lincoln Laboratory, who studied the propagation of a CW infinite Gaussian with a e^{-2} diameter of 70 cm. The absorption coefficient was 0.07 km^{-1}, with no scattering. The laser was twice-diffraction-limited DF with a wavenumber of 8.5×10^3 cm^{-1}. Two cases were considered at focal lengths of 2, 5, and 10 km. The first case had a power of 10 MW, a wind speed of 250 m/s, and no slewing. The second case had 2 MW power, a 2-m/s wind, and a 0.02-s^{-1} slew. The results, consisting of the area containing 63% of the focal-plane power and of the peak intensity are summarized in Table 4. A_{rel} and I_{rel} compare these quantities with those that would have been obtained if there were no thermal blooming. The results for these highly bloomed cases agree within about 5% with those of Herrmann.

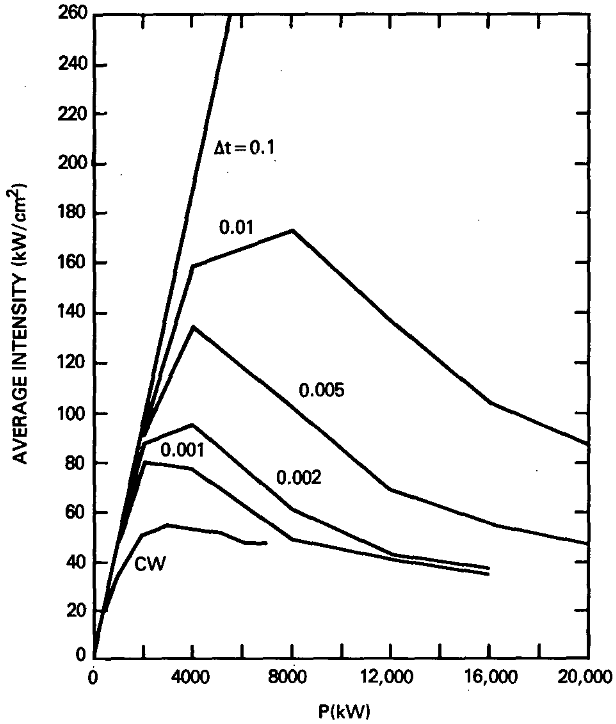

Fig. 2—SSPARAMA results (F = 1 km, diam = 70.7 cm (1/e), α = 0.1 km^{-1}, k = 2966 cm^{-1}, v_0 = 10 m/s, and Ω = 0.1)

Table 4—SSPARAMA Results for the Propagation of a CW Infinite Gaussian With a Wavenumber of 8500 cm^{-1}, an e^{-2} Diameter of 70 cm, an Absorption Coefficient of 0.07 km^{-1}, and No Scattering

Focal Length F (km)	Area A Containing 63% of the Focal-Plane Power (cm^2)	Relative Area A_{rel} Relative To No Thermal Blooming	Peak Intensity I_{peak} (kW/cm^2)	Relative Peak Intensity I_{rel} Relative To No Thermal Blooming
First Case: 10 MW Power, 250-m/s Wind, and No Slew				
2	57.6	20.3	147	0.0464
5	658	37.0	10.3	0.0251
10	3543	49.8	1.33	0.0184
Second Case: 2 MW Power, 2-m/s Wind, and 0.02-s^{-1} Slew				
2	64.8	22.8	26.8	0.0422
5	474	26.6	2.96	0.0359
10	2018	28.4	0.495	0.0341

Another example of SSPARAMA output is illustrated in Fig. 3, namely, the final contour plot for the 5-km run from the first case with 10% contour levels. The complete printed output from SSPARAMA is included in Figs. 4a through 4c.

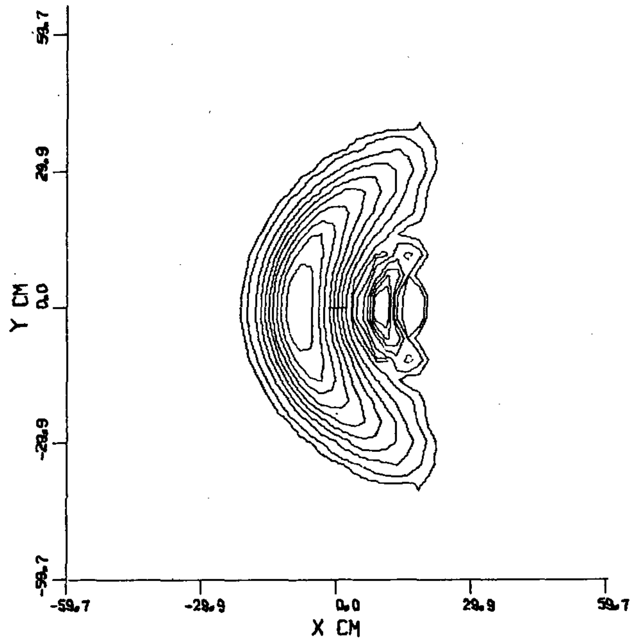

Fig. 3—Contour plot with 10% contour levels for the 5-km run from the first case in Table 4 (PNALF = 0.350, PNK = 10.400, PNO = 0.002, PNS = 0.000, PND = 80.000)

Figure 4a, the first page of printed output, is almost self-explanatory. Both dimensionless and physical parameters are listed; one is computed from the other, depending on which was entered. The program options indicate the mode, either CW or MP and the beamshape etc. The results summary in Fig. 4a includes the final value of the energy conservation integral, Eq. (2). This quantity, which is ideally equal to 1, gives a quick check on the validity of the numerical calculations. One factor that limits the accuracy is the use of a finite mesh size. As this mesh is made finer, the intensity distribution gets closer to the mesh boundaries, and numerical errors may enter through diffraction and the use of a discrete Fourier-transform routine as energy is reflected off the boundary. To avoid this reflection, the outermost boundary of the computational grid is set to zero and the next outermost boundary is set to one half its value at each z step. Thus the sum over normalized intensity gives an indication of how much energy was lost due to boundary-value problems.

The area that is given in Fig. 4a is the area containing exactly 0.63 of the total flux obtained by linear interpolation between adjacent flux fractional areas. This area will include contributions from several peaks as the intensity pattern breaks up under severe blooming conditions, so its meaning may also require a suitable interpretation of the intensity contour map. In addition the relative area and maximum intensity are calculated relative to the focal area and intensity of a vacuum-propagated infinite Gaussian whose e^{-1} diameter is equal to the value of WIDTH regardless of the beamshape being propagated.

```
*** MEPHISTO INPUT DATA ***

DIMENSIONLESS PARAMETERS      PHYSICAL PARAMETERS              NUMERICAL PARAMETERS

  NA  =  0.000049             RADIUS(M)  =      0.248            HX  = 0.20
NALF  =     0.350          ALPHA(1/KM)  =   0.070000            HY  = 0.20
  NK  =     10.40             K(1/CM)   =    8488.93            NX  =   64
  NC  =      0.00             V(M/S)    =     249.98            NY  =   64
  NS  =      0.00       OMEGA(RAD/SEC)  =     0.0000         PHIMXX =1.000
  ND  =     80.00          ENERGY(KJ)   =   10420.69
  NZ  =    1.0000             F(KM)     =      5.000
                             DT(SEC)    =   1.000000
                         ALPHAS(1/KM)   =   0.000000

PROGRAM OPTIONS

                      MODE    CW
                 BEAMSHAPE    INFINITE GAUSSIAN
                  ADAPTION    YES
   HALF-STEP INTEGRATION     YES
        PUNCHED CARD OUTPUT   NO
           NUMBER OF PLOTS     5
        LOW LEVEL CONTOURS    NO
  RESCALE FINAL CONTOUR PLOT  YES

                                  *** RESULTS ***

                      THE CALCULATIONS REACHED Z =   5.00000 (KM)
                THE SUM OVER NORMALIZED INTENSITY =   1.00000
                           THE NUMBER OF Z-STEPS =   22
      AVERAGE POWER (KW) EMITTED AT APERTURE = 10420.694
            AVERAGE TRANSMITTED POWER (KW) =  7343.339
      AREA (SQCM) CONTAINING 0.63 OF POWER =   657.995
          A REL (RELATIVE TO INF. GAUSSIAN) =    36.982
   AVERAGE INTENSITY (KW/SQCM) IN THIS AREA =     7.031
                 PEAK INTENSITY (KW/SQCM) =    10.345
       I REL (RELATIVE TO INF. GAUSSIAN PEAK) =   0.02507
```

Fig. 4a—First page of the output by SSPARAMA, containing the input that resulted in Fig. 3 and a summary of the results

Figure 4b, the page containing numerical data, begins with a list of internally computed quantities that relate to the problems of air breakdown and t-cubed self-blooming. They are printed only for possible future data analysis. Assuming the breakdown intensity at 10.6 μm is 3×10^6 W/cm^2 and that this is inversely proportional to wavelength squared, the following quantities are computed as a function of range: the minimum area required for breakdown, the ratio of this minimum area to the vacuum area, the maximum pulselength before breakdown occurs, the critical power, the saturation time, the intensity produced by the critical power propagating in a vacuum, and factors accounting for turbulence with values of C_n^2 of 10^{-15} and 10^{-14}. This is followed by an x and y slice through the aperture to check the initial beamshape.

The quantities, including the values of HZN in z/ka^2 units, relating to the coordinate system adaption are printed at each z step. The headings D, D1, D2, ALPHA1, ALPHA2, BETA1, DALPH1, DALPH2, DBET1, and XCEN correspond to D, D_1, D_2, α_1, α_2, β, $\Delta\alpha_1$, $\Delta\alpha_2$, $\Delta\beta$, and X used in the second section of this report. Also included is EPSMX, the maximum value of the summation given in Eq. (72); PHIMX, the maximum value of the positive phase change applied to ψ to obtain Φ; and PARM, the number of pulses, for the MP mode, that occur in a computational cell.

Figure 4c, the output data, lists in the top portion the area, flux, the area fraction, and flux fraction contained within each contour level. From these data the 63% area is interpolated. This is followed in the middle portion by the z locations of the maximum of the average and peak intensities, the minimum 63% area, and the minimum z step that

*** NUMERICAL DATA ***

RANGE(KM)	AS2(CM2)	AS2/A2D	TP(SEC)	PCR(W)	ISAT(SEC)	ISAT(W/CM2)	TURBCOR1	TURBCOR2
0.05000	98.99318	0.16193	5.4439-003	1.0429+005	5.4483-005	4.0712+001	1.0000+000	1.0000+000
0.56056	1018.03493	2.10781	5.0917-004	5.1276+005	2.5054-005	2.4370+002	9.9997-001	9.9947-001
1.11611	1395.21180	3.77194	3.5735-004	6.5774+005	2.2555-005	3.9261+002	9.9961-001	9.9378-001
1.67167	1658.41931	6.09563	2.8917-004	7.6732+005	2.1293-005	5.9895+002	9.9805-001	9.6992-001
2.22722	1859.23530	9.81094	2.4810-004	8.6091+005	2.0497-005	9.2798+002	9.9301-001	8.9967-001
2.78278	2018.48219	16.51645	2.1981-004	9.4546+005	1.9943-005	1.5200+003	9.7823-001	7.3929-001
3.33833	2147.25482	30.59665	1.9875-004	1.0243+006	1.9536-005	2.7584+003	9.3513-001	4.7630-001
3.89389	2252.39000	67.41011	1.8224-004	1.0994+006	1.9227-005	5.9811+003	8.0745-001	2.0923-001
4.44944	2338.50222	196.31625	1.6884-004	1.1720+006	1.8988-005	1.7202+004	4.9384-001	5.7989-002
5.00500	2408.90687	424.44483	1.5765-004	1.2428+006	1.8801-005	3.6826+004	2.4185-001	1.9731-002

NORMALIZED AMPLITUDE SAMPLES AT APERTURE

AT X= 32 Y= 1 TO 64

0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.1 0.1 0.1 0.2 0.2 0.3 0.4 0.4 0.5 0.5 0.6
0.6 0.5 0.5 0.4 0.4 0.3 0.2 0.2 0.1 0.1 0.1 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

AT Y= 32 X= 1 TO 64

0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.1 0.1 0.1 0.2 0.2 0.3 0.4 0.4 0.5 0.5 0.6
0.6 0.5 0.5 0.4 0.4 0.3 0.2 0.2 0.1 0.1 0.1 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0

Z	HZ	D	D1	D2	ALPHA1	ALPHA2	BETA1	DALPH1	DALPH2	DBET1	XCEN	EPSMX	PHIMX	PARM
0.224	2.2-003	0.9124	0.9124	0.9124	-4.9649	-4.9649	0.0000	0.0001	0.1749	-0.2820	0.0000	5.5-004	0.0000	2.0-004
0.442	4.2-003	0.8311	0.8311	0.8339	-4.7360	-4.5769	-0.2692	0.0009	0.1781	-0.2833	-0.0306	5.2-004	0.4949	1.9-004
0.655	4.1-003	0.7555	0.7555	0.7685	-4.5120	-4.2066	-0.5267	0.0024	0.1808	-0.2839	-0.0920	5.0-004	0.4935	1.8-004
0.863	4.0-003	0.6849	0.6849	0.7004	-4.2912	-3.8530	-0.7719	0.0049	0.1857	-0.2851	-0.1845	4.7-004	0.4919	1.7-004
1.068	4.0-003	0.6189	0.6190	0.6458	-4.0719	-3.5131	-1.0048	0.0086	0.1934	-0.2870	-0.3091	4.4-004	0.4903	1.6-004
1.271	3.9-003	0.5570	0.5574	0.5931	-3.8525	-3.1833	-1.2258	0.0136	0.2043	-0.2898	-0.4673	4.1-004	0.4885	1.6-004
1.471	3.9-003	0.4988	0.4996	0.5480	-3.6310	-2.8601	-1.4350	0.0204	0.2194	-0.2937	-0.6615	3.9-004	0.4867	1.5-004
1.672	3.9-003	0.4441	0.4456	0.5081	-3.4058	-2.5392	-1.6325	0.0295	0.2396	-0.2990	-0.8946	3.6-004	0.4850	1.4-004
1.872	3.9-003	0.3927	0.3951	0.4733	-3.1748	-2.2156	-1.8189	0.0414	0.2664	-0.3062	-1.1706	3.3-004	0.4832	1.3-004
2.073	3.9-003	0.3442	0.3481	0.4436	-2.9360	-1.8830	-1.9947	0.0572	0.3017	-0.3161	-1.4946	3.0-004	0.4815	1.2-004
2.277	3.9-003	0.2986	0.3046	0.4193	-2.6871	-1.5328	-2.1615	0.0780	0.3478	-0.3301	-1.8735	2.8-004	0.4799	1.2-004
2.484	4.0-003	0.2554	0.2644	0.4006	-2.4240	-1.1535	-2.3214	0.1067	0.4101	-0.3538	-2.3199	2.5-004	0.4744	1.1-004
2.696	4.0-003	0.2150	0.2281	0.3886	-2.1443	-0.7271	-2.4849	0.1445	0.4853	-0.3864	-2.8435	2.3-004	0.4772	1.0-004
2.912	4.1-003	0.1775	0.1961	0.3847	-1.8449	-0.2552	-2.6624	0.1945	0.5707	-0.4370	-3.4622	2.1-004	0.4764	9.5-005
3.134	4.2-003	0.1429	0.1690	0.3905	-1.5209	0.3535	-2.8774	0.2579	0.6491	-0.5090	-4.2020	1.9-004	0.4761	8.8-005
3.367	4.4-003	0.1109	0.1473	0.4084	-1.1654	1.0209	-3.1616	0.3250	0.6899	-0.5920	-5.1128	1.7-004	0.4705	8.1-005
3.610	4.6-003	0.0821	0.1321	0.4412	-0.7796	1.7837	-3.5548	0.3473	0.6543	-0.6258	-6.2478	1.6-004	0.4774	7.6-005
3.869	4.8-003	0.0567	0.1239	0.4911	-0.3994	2.5777	-4.0476	0.3004	0.5712	-0.5733	-7.6632	1.5-004	0.4793	7.2-005
4.142	5.1-003	0.0358	0.1220	0.5595	-0.0769	3.3661	-4.5864	0.2350	0.4881	-0.4779	-9.3684	1.5-004	0.4819	7.0-005
4.431	5.4-003	0.0202	0.1258	0.6484	0.1831	4.1540	-5.1423	0.1912	0.4283	-0.4030	-11.3604	1.5-004	0.4852	6.9-005
4.740	5.7-003	0.0110	0.1352	0.7622	0.4112	4.9725	-5.7477	0.1636	0.3889	-0.3554	-13.6639	1.6-004	0.4884	7.0-005
5.000	6.2-003	0.0092	0.1471	0.8731	0.6177	5.7417	-6.3666	0.1158	0.2889	-0.2570	-15.7174	1.7-004	0.4909	7.3-005

Fig. 4b—Second page of the output, containing numerical data

have occurred during beam propagation. Then in the lower portion the peak and average intensities, the 63% area and the location of the peak intensity in centimeters are listed at each z step.

Summary of Program Structure

When the half-step integrations are used, the solution is advanced twice before the information at each z step is stored. This can be seen from the flow chart of SSPARAMA (Fig. 5). Thus, when NMS = 1, the program actually used twice the number of z steps that are printed and included points approximately midway between those listed.

The structure of the code SSPARAMA is explained below and summarized in the flow chart in Fig. 5.

- The call to subroutine START causes the input data to be read. The real part of the 64-by-64 array ψ is defined according to the beamshape specified. Initially the phase of this array is zero.

*** OUTPUT DATA ***

LEVEL	AREA (SQ CM)	FLUX (KW)	AREA FRACTION	FLUX FRACTION	IRRADIANCE (KW/SQ-CM)
0.9000	8.781+001	8.615+002	0.0556	0.1173	9.811+000
0.8000	1.756+002	1.630+003	0.1111	0.2219	9.280+000
0.7000	2.634+002	2.320+003	0.1667	0.3159	8.806+000
0.6000	4.039+002	3.274+003	0.2556	0.4458	8.105+000
0.5000	5.795+002	4.261+003	0.3667	0.5803	7.353+000
0.4000	7.200+002	4.915+003	0.4556	0.6693	6.826+000
0.3000	9.483+002	5.754+003	0.6000	0.7836	6.067+000
0.2000	1.177+003	6.323+003	0.7444	0.8611	5.374+000
0.1000	1.581+003	6.930+003	1.0000	0.9438	4.383+000
0.0500	1.826+003	7.126+003	1.1556	0.9704	3.902+000

MAXIMUM AVG I = 1.873+001 AT Z= 3.367+000
MAXIMUM PEAK I= 3.841+001 AT Z= 3.610+000
MINIMUM AREA = 2.741+002 AT Z= 3.610+000
MINIMUM HZ = 2.170-003 AT Z= 1.000-006

Z	IAVE	A63	IMAX	XPEAK	YPEAK
0.000	3.421	1918.802	5.308	0.000	0.000
0.224	3.688	1752.298	5.719	0.000	0.000
0.442	3.973	1602.249	6.152	0.000	0.000
0.655	4.277	1466.170	6.608	0.000	0.000
0.863	4.604	1342.439	7.083	0.000	4.143
1.068	4.970	1225.665	7.570	0.000	3.972
1.271	5.361	1120.431	8.060	0.000	3.812
1.471	5.792	1022.527	8.533	0.000	3.664
1.672	6.274	930.908	9.217	-3.304	3.528
1.872	6.791	848.033	9.930	-3.112	0.000
2.073	7.430	764.166	10.776	-5.841	3.297
2.277	8.260	677.725	12.064	-5.464	3.205
2.484	9.320	591.981	13.519	-7.636	3.133
2.696	10.786	503.964	16.524	-7.092	3.086
2.912	12.922	414.367	20.387	-6.576	6.140
3.134	15.727	335.210	25.611	-6.105	6.186
3.367	18.727	276.968	32.220	-5.700	6.327
3.610	18.601	274.121	38.413	-5.398	6.576
3.869	15.599	321.008	33.563	-5.227	3.469
4.142	12.970	378.764	21.360	-3.458	3.703
4.431	10.239	470.186	16.150	-5.267	0.000
4.740	8.277	569.200	12.452	-5.460	0.000
5.000	7.031	657.995	10.345	-5.695	0.000

Fig. 4c—Third page of the output, containing the remaining numerical data

- The initialization procedure continues with the call to INTENS, where the aperture intensity is computed at each mesh point.

- The call to DENS computes the quantity $g(x, y, z)$ given in Eq. (63) and then applies the phase change given by Eq. (62) which converts ψ to Φ. The first z increment is also computed.

- The main program loop begins here with a call to OUTPUT to store various values until the calculations are completed.

- The call to ADVANCE applies the Fourier transform of Eq. (67) and then the phase change of Eq. (68). The array is Fourier-transformed back to yield $\Phi(z + \Delta z)$.

- The intensity is computed with the call to INTENS, and the boundary values of the array are tapered to zero.

- The call to DENS now includes a call to VTRANS, by which the phase change of Eq. (62) is reversed, converting Φ back to ψ. The quantities $\{\alpha_1, \alpha_2, \beta\}$ and $\{\Delta\alpha_1, \Delta\alpha_2, \Delta\beta\}$ are found in VTRANS, and the values of D_1 and D_2 are updated. After the return to DENS, Eq. (63) is solved and the phase change of Eq. (62) is reapplied, converting ψ back to Φ in preparation for the next call to ADVANCE.

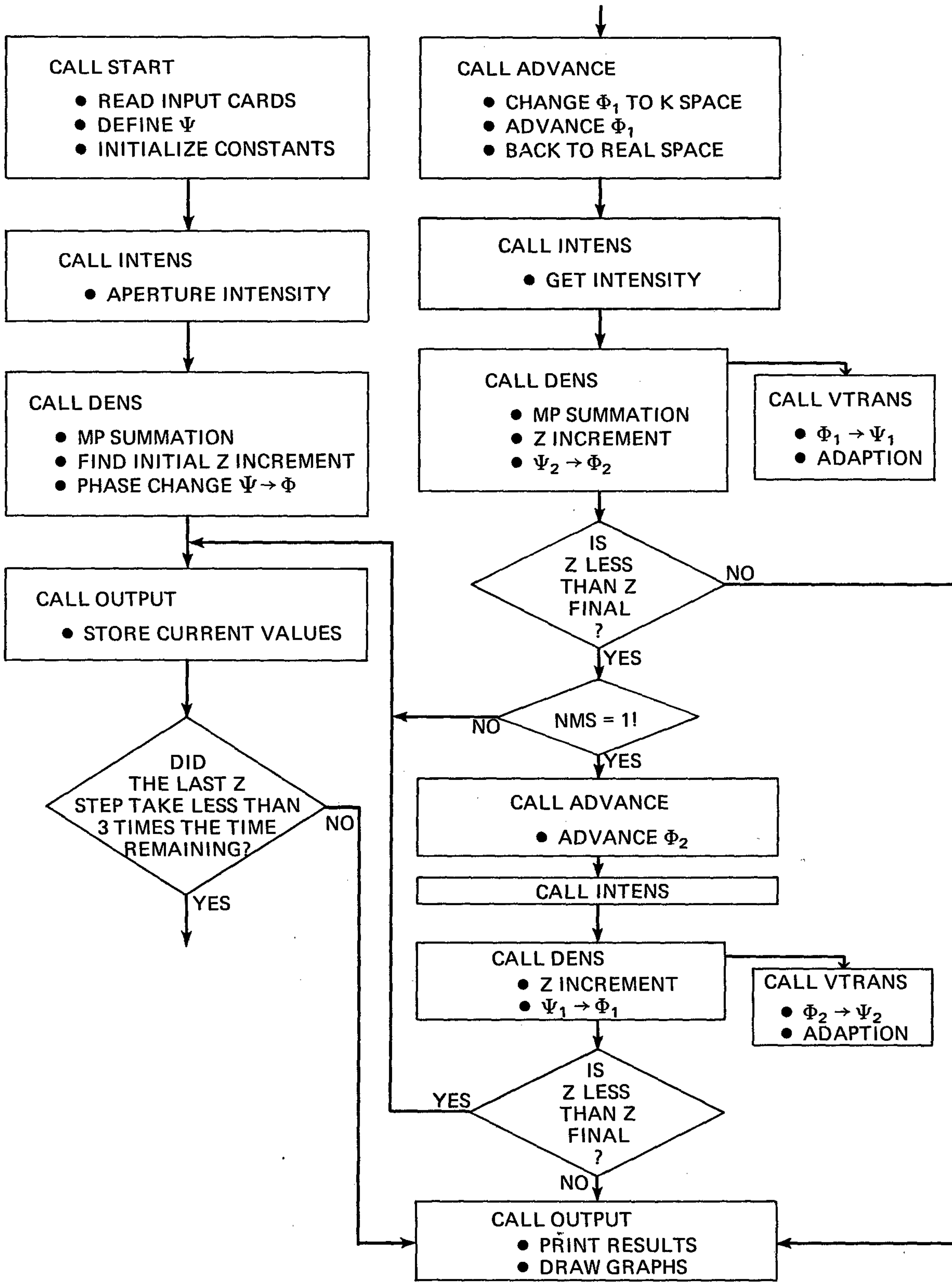

Fig. 5—Summary of the code SSPARAMA

- Now that one cycle of propagating the solution is completed, the code checks if z final has been reached and if the half-step integrations are to be performed as outlined in the section titled Numerical Procedures.

- When z final has been reached or the time limit of execution is near, the last call to OUTPUT prints the results and ends this run.

The Appendix contains a complete listing of the code with copious comments included.

REFERENCES

1. P.B. Ulrich, J.N. Hayes, J.H. Hancock, and J.T. Ulrich, "Documentation of PROP E, a Computer Program for the Propagation of High Power Laser Beams Through Absorbing Media," NRL Report 7681, May 29, 1974.
2. P.B. Ulrich, "PROP-I: An Efficient Implicit Algorithm for Calculating Nonlinear Scalar Wave Propagation in the Fresnel Approximation," NRL Report 7706, May 29, 1974.
3. A.H. Aitken, J.N. Hayes, and P.B. Ulrich, "Propagation of High-Energy 10.6-Micron Laser Beams Through the Atmosphere," NRL Report 7293, May 28, 1971.
4. J. Wallace and J.Q. Lilly, "Thermal blooming of repetitively pulsed laser beams," *J. Opt. Soc. Am.* 64, 1651–1655 (Dec. 1974).
5. J. Herrmann and L.C. Bradley, "Numerical Calculation of Light Propagation," Massachusetts Institute of Technology, Lincoln Laboratories, Lexington, Mass. Report LTP-10, July 12, 1971.
6. H.J. Breaux, "An Analysis of Mathematical Transformations and a Comparison of Numerical Techniques for Computation of High Energy CW Laser Propagation in an Inhomogeneous Medium," BRL Report 1723, June 1974.
7. K.G. Whitney and P.B. Ulrich, "Scaling Laws for Multipulse Steady-State Thermal Blooming," NRL Memorandum Report 3229, Mar. 1976, Appendix B3.

APPENDIX A
Listing of Code and Comments

```
      PROGRAM SSPARAMA
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      COMMON /999/ A1(64,64), A2(64,64), ITENS(64,64)
     COMMON /AAA/ EPS(64,64),EPSO(64,64),AOUT(11,99),BOUT(9,10),
    . AIN(2,64),DALPH1(2),DALPH2(2),DBET1(2),ALPH10(2),ALPH20(2),
    .   BET10(2), D10(2), D20(2), RD10(2), RD20(2), SRTD10(2), XCENO(2)
      COMMON /BBB/ TENS(64,64), G1(64), G2(64), PHASE1(64), PHASE2(64),
    . CONMIN(10),M2(3),SV1(64),SV2(64),PARM(80)
      COMMON /SINGLS/ F, PNA, PNALF, PNK, PNO, PNS, PND, PNZ, HX, HY,
    .   HZ, Z, ZZ,ZZF, ZNM, ZFINAL, XZERO, YZERO, WIDTH, ALPHA, WN,
    .   VODT, OMDT, HT, ENERGY, ALPHAC, CS, REFRAC, GAMMA, ETJ, CTK,
    .  EJTKJ, RHT, POUT, DAREA, W2, TS, TPULSE, AS2,PCR,SI,TCOR1,TCOR2,
    .   Z1, RI63MX, Z2, RIMXMX, Z3, APMN, Z4, HZMN, DKAREA, TENSMX,
    .   EX, PHIMX, EPSMX, ERRMX, DGMX, R1, BDIMAX, VTERM, PHIMXX,HZNMS,          14
    .   PI, IMAX, JMAX, NX, NY, NAD, NX2, NY2, NXY, NXDIM, NYDIM, NPT,
    .   IPLOT,NITER,NBUF,NXM,NYM,NMS,NFLAG,D,D1,D2,P1,P2,SRTD1,SRTD2,
    .   RSRD12,XCEN,TLAST,SQRT8,PND0,GCON0,GCON,BDI,HCZ10,HCZ20,HCZ1N,
    .   HCZ2N,HCZ12,ALPH1,ALPH2,BET1,CON1,CON2,HZ0,HZN,EXO,EXN,WT1,WT2
     COMMON /OUTS/ NBM,SCLFAC,NRS,NPM,NCW,NEXIT,NPLOT,NPUNCH
C* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      COMPLEX A1, A2
      LOGICAL LS
      DATA (CS=34000.0), (REFRAC=0.154), (GAMMA=1.4), (ETJ=1.0E-7),
    .   (CTK=1.0E-5),  (PI=3.14159265), (BDI=3.0E6), (EJTKJ=1.0E-3)
      DATA (NXDIM=64), (NYDIM=64), (NZ=20)
      BANK,(0),/999/
C
C     INPUT AND INITIALIZATION
C
     TSTART=TIMELEFT(DUMMY)
      LS=.FALSE.
55   CONTINUE
     CALL START(LS)
     NEXIT=0
      NITER=0
      IPLOT=0
      ZZ1=0.0
      ZZ2=0.0
      I2=2
      IF (NMS .EQ. 0) I2=1
C
      CALL INTENS(A1,.FALSE.)
      CALL DENS(A1, A1, ZZ1, 1, 1, .FALSE.)
      HZO=0.0
C
C*************************************************************************
C
C   MAIN PROGRAM LOOP
C
14    CONTINUE                                                                 C
      NITER=NITER+1
C
C   STORE VALUES FOR LATER PRINTOUT
C
2      CALL OUTPUT(.FALSE.)
C
C   IF TIME REMAINING IS LESS THAN 3 TIMES THAT FOR THE LAST
C   Z STEP - EXIT
C
```

```
   3    TNOW=TIMELEFT(0)
        DT=TLAST-TNOW
        TLAST=TNOW
        IF(3*DT.LE.TNOW) GO TO 8
        PRINT 22,Z
 22     FORMAT(//25X25H*** TIME ABORT AT Z(KM) =F10.5,1X3H***//)
        GO TO 13
    8   CONTINUE
C
C       ADVANCE FROM ZZ TO ZZ+DZ, CALCULATING NEW AMPLITUDES IN A
C
 40      CALL ADVNCE(A1,1)
         CALL INTENS(A1, .FALSE.)
         CALL DENS(A1, A2, ZZ1, 1, I2, .TRUE.)
         IF (Z .GE. ZFINAL) GO TO 15
C                                                                       C
C    REPEAT IF HALF-STEP INTEGRATION IS INCLUDED
C
         IF (NMS .EQ. 0) GO TO 45
         CALL ADVNCE(A2,2)
         CALL INTENS(A2, .FALSE.)
         CALL DENS(A2, A1, ZZ2, 2, 1, .TRUE.)
         IF (Z .GE. ZFINAL) GO TO 15
 45     CONTINUE
         GO TO 14
C
C*************************************************************************************
C
C    SET NEXIT EQUAL 1 FOR PREMATURE EXITS
C
 13     NEXIT=1
 15     CONTINUE
C
C    EXECUTE ALL OUTPUT
C
        CALL OUTPUT(.TRUE.)
        PRINT 16
  16    FORMAT(1H1)
  17    CALL STOPPLOT

C           PRINT RUN TIME (CP TIME).
        TRUN=(TSTART-TIMELEFT(DUMMY))/60.
        PRINT 18,TRUN
   18 FORMAT(//,16(1H*),*  RUN TIME=*,F6.2,* MINUTES*)

        STOP
        END
```